UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

TRABAJO FIN DE MÁSTER

DETECCIÓN DE HABLANTES EN LOCUCIONES CORTAS EN AUDIO BROADCAST

Máster Universitario en Ingeniería de Telecomunicación

Autor: Álvaro Escudero Barrero

Tutor: Joaquín González Rodríguez
Dpto. de Tecnología Electrónica y de las Comunicaciones

FECHA: Febrero 2018

DETECCIÓN DE HABLANTES EN LOCUCIONES CORTAS EN AUDIO BROADCAST

AUTOR: Álvaro Escudero Barrero
DIRECTOR: Joaquín González Rodríguez

AUDIAS - Audio, Data Intelligence and Speech
Dpto. de Tecnología Electrónica y de las Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Febrero 2018

Resumen

Resumen

En este trabajo de fin de máster se estudia, desarrolla y optimiza un sistema de detección de hablantes en locuciones cortas y audio broadcast. Para ello, se hará uso de técnicas del estado del arte en reconocimiento de locutor, mejoras para la detección de locuciones cortas y técnicas de adaptación de dominio en audio broadcast.

Se implementará un sistema completo de reconocimiento automático de locutor basado en i-vectors. Además, dada la problemática inherente a los sistemas de reconocimiento de locutor en audio broadcast (radio y TV), que deben lidiar con apariciones de corta duración (declaraciones, entrevistas, tertulias, etc.) y además, deben trabajar en entornos acústicos y de grabación nada homogéneos (ruedas de prensa, mítines, conexiones telefónicas, etc.), se proponen alternativas que permitan mejorar el rendimiento del sistema adaptándose a estas características.

Dicho sistema será evaluado de forma que permita comparar las distintas versiones desarrolladas, con el objetivo de realizar un estudio de aquellas técnicas propuestas que permiten mejorar en el rendimiento del sistema. Por tanto, estas pruebas se realizan sobre un entorno experimental con datos extraídos de programas radiofónicos (Audias-Radio-2015) que se adaptan adecuadamente a la tarea.

Finalmente, se analizan los resultados obtenidos en términos de tasa de error (EER) y se presentan las conclusiones extraídas a partir de los mismos.

Palabras Clave

Detección de locutor, reconocimiento automático de locutor, locuciones cortas, adaptación de dominio, audio broadcast, i-vector, reconocimiento de patrones.

Abstract

This master thesis is focused on study, develop, and optimice an automatic speaker recognition system for detecting speakers in short utterances and broadcast audio. In order to do this, state-of-the-art techniques will be used for speaker recognition. In addition, we propose some improvements for the detection of short utterances and domain adaptation techniques for broadcast audio.

A complete system of automatic speaker recognition based on i-vectors will be implemented. Given the inherent problems of speaker recognition systems in broadcast audio (radio and TV), -which must deal with short-term appearances (statements, interviews, gatherings, etc.) and also, they must work in acoustic environments and of non-homogeneous recording (press conferences, meetings, telephone connections, etc.)-.

This system will be evaluated to compare the different developed versions, with the aim of carrying out a study of those proposed techniques that allow improving the performance of the system. Therefore, these tests are performed on an experimental environment with data extracted from radiophonic programs (Audias-Radio-2015) that adapt appropriately to the task.

Finally, the results obtained are analyzed in terms of equal error rate (EER) and the conclusions are presented.

Key words

Speaker recognition, short utterances, domain adaptation, broadcast audio, i-vectors, pattern recognition.

A MIS PADRES.
A MI HERMANO.

A ANDREA.

Agradecimientos

En primer lugar, me gustaría agradecer a Joaquín la oportunidad que me ha dado de realizar este trabajo de fin de máster, bajo su supervisión y sus consejos, que de tanto han servido para llevar este proyecto a buen término. Además, me gustaría aprovechar estas líneas para agradecer su confianza depositada en mí y darme la oportunidad de formar parte de Audias. Ha sido una experiencia enormemente enriquecedora. Gracias.

Han sido dos años compartiendo momentos en el C-109, con grandes investigadores y profesores pero sobre todo compañeros. Joaquín, Doroteo, Daniel, Javier F., Alicia, Rubén, Adrián, Pilar, Sara, Marcos, Carlos, Beltrán, Diego y por supuesto a los vecinos Rubén T. Rubén V. Aythami, Ester, Javier, Alejandro, etc. Espero no dejarme a nadie en el tintero...

Por supuesto, también quiero agradecer a mis compañeros Raúl y José Antonio. Tras tantas horas de esfuerzo, largas tardes de estudio, prácticas hasta medianoche y sobre todos las risas en el C005-1. Gracias amigos.

Gracias a las personas más importantes de mi vida: mis padres. Ellos me han enseñado todo. Soy quien soy gracias a vosotros. Os quiero.

Gracias a mi hermano, que siempre ha sido un ejemplo a seguir y lo seguirá siendo por muchos años que pasen y nos hagamos mayores. Te quiero.

Gracias a Andrea, por apoyarme durante este largo camino y estar día tras día a mi lado para hacerme sonreír. Te quiero.

Álvaro.

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción

La tarea de detección de hablantes en locuciones cortas continúa siendo una consideración clave al implementar un sistema de reconocimiento automático de locutor, debido a la multitud de aplicaciones en el mundo real que a menudo tienen acceso a datos de voz de duración limitada. Una de estas aplicaciones es la detección de hablantes en audio *broadcast* (radio y TV). Caracterizado habitualmente por apariciones muy cortas (declaraciones, entrevistas, noticias, etc.) y en entornos acústicos y de grabación nada homogéneos (ruedas de prensa, pasillos de congreso, mítines, conexiones telefónicas, etc.).

1.1. Motivación del proyecto

El reconocimiento de locutor es una área altamente consolidada en los últimos años debido a un alto rendimiento de los sistemas automáticos [Reynolds et al., 2017], cuando se dispone de suficiente duración y canales homogéneos [Sarkar et al., 2012]. Por tanto, el rendimiento asociado a estos sistemas decrece significativamente cuando las condiciones del entorno no se encuentran controladas [Domínguez et al., 2012].

Habitualmente los sistemas de reconocimiento automático de locutor que se implementan en aplicaciones reales deben lidiar con esta problemática, donde la duración de las locuciones puede ser muy corta (1-10 segundos) y los entornos de grabación pueden contener alta variabilidad (ruido, distorsión, etc).

Una de estas aplicaciones se refiere al uso de sistemas automáticos de detección de locutores en audio *broadcast* (radio y TV). En este caso, los sistemas deben ser capaces de trabajar con un flujo continuo de audio (noticiario, *magazine*, concurso, tertulia, entrevista, etc.) donde aparecen multitud de locutores, de forma repetida o esporádica. Cada uno de estos locutores pueden intervenir con locuciones de diferente duración (desde segundos, hasta decenas de minutos) y además, se pueden presentar multitud situaciones que produzcan un entorno de alta variabilidad, como:

- Dispositivos y calidad de grabación: micrófono de estudio, conexión telefónica, grabadora de audio, etc.
- Entorno acústico: plató/estudio de grabación, corresponsal en exteriores, sintonía de programa, efectos de sonido, etc.
- Intervención de locutores: solape entre locutores, intervenciones incompletas, música de fondo, interjecciones, etc.

Todo ello, compone un escenario de trabajo complejo. Donde se debe trabajar con una problemática inherente a las características del audio, teniendo como objetivo obtener la menor degradación posible en el rendimiento del sistema.

Además, las técnicas actualmente utilizadas en el desarrollo de sistemas de reconocimiento automático de locutor son altamente dependientes del acceso de gran cantidad de datos etiquetados (es necesario conocer que locuciones corresponden a cada locutor de manera fiable) y no etiquetados. Habitualmente, son necesarias decenas de locuciones de cientos de locutores, con el objetivo de construir sistemas suficientemente robustos que proporcionen un alto rendimiento. Dichos datos deben provenir de un entorno, denominado "dominio", similar al tipo de datos sobre el que está orientada la aplicación en concreto. Por ejemplo, un sistema de reconocimiento sobre llamadas telefónicas en inglés, tomará gran cantidad de audio telefónico en inglés, con el objetivo de ser más robusto sobre esa tarea específica.

Sin embargo, se antoja poco realista esperar disponer gran cantidad de datos etiquetados, que además presenten las mismas condiciones (acústica, calidad, idioma, etc.) cuando se diseña un sistema de reconocimiento automático de locutor para una aplicación concreta.

En el caso de un sistema de detección de locutores aplicado a audio *broadcast*, debería ser entrenado con gran cantidad de datos etiquetados provenientes de un dominio similar (programas de radio y TV). Esto conllevaría un alto coste en recursos (humanos, temporales y económicos). Sin embargo, y a partir de grandes esfuerzos en investigación, existen técnicas que tratan de adaptar sistemas específicamente diseñados para un dominio específico (*domain adaptation*), que han sido entrenados con miles de horas de audio (ej: habla telefónica en inglés) a partir de una pequeña cantidad de datos (cientos de horas de audio) provenientes de otro dominio (ej: habla microfónica en castellano). Esto permite construir sistemas específicos mucho más fiables y utilizando la menor cantidad de datos posible.

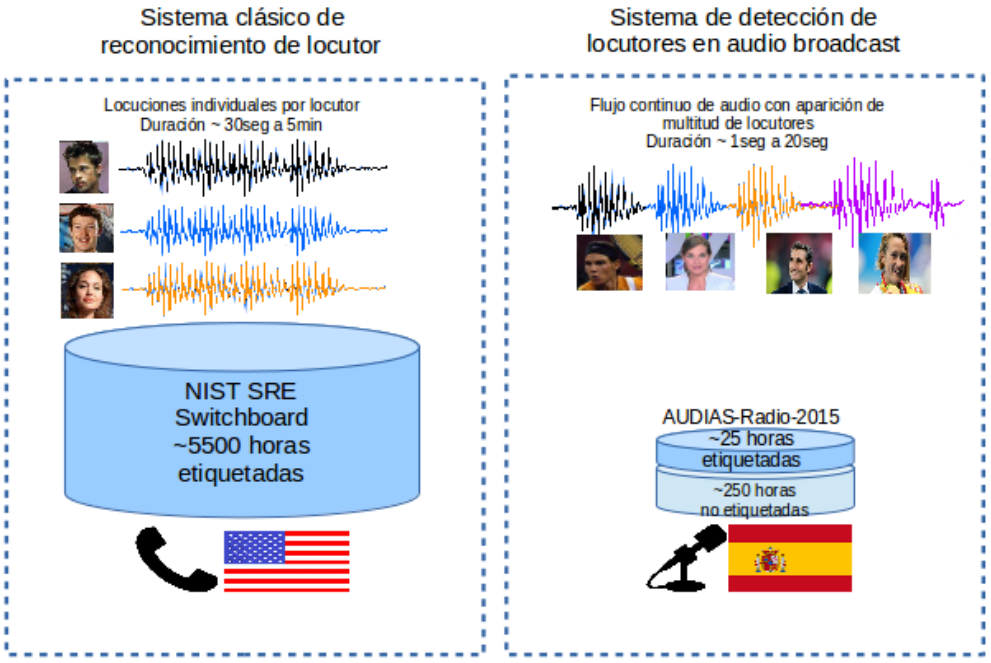

Figura 1.1: Esquema comparativo entre problemática de sistema clásico (baseline) y sistema de locuciones cortas en audio broadcast (objetivo).

Por estos motivos, este proyecto tratará de diseñar, desarrollar e implementar un sistema de detección de locutores capaz de trabajar con un flujo continuo de audio, donde existen gran cantidad de locuciones cortas (alrededor de 5 segundos) proveniente de una fuente de audio *broadcast* (radio y TV) con alta variabilidad. Para ello, se hará uso de una base de datos en el dominio de la aplicación (*in-domain*) con alrededor de 250 horas (25 etiquetadas) de audio (habla microfónica en castellano), denominada Audias-Radio-2015 y otra base de datos con gran cantidad de audio (aproximadamente 5500 horas) fuera del dominio (*out-domain*), principalmente proveniente de habla telefónica en inglés, ver Figura 1.1.

1.2. Objetivos y planteamiento

El principal objetivo de este proyecto es el desarrollo, estudio y optimización de un sistema end-to-end de detección de hablantes, específico para un entorno de locuciones cortas y audio broadcast (radio y TV). Para ello, se proponen los siguientes objetivos secundarios:

- Estudio de las diferentes técnicas utilizadas en el estado del arte en reconocimiento de locutor.
- Implementación de un sistema de reconocimiento de locutor basado en i-vectors.
  - Adaptación del sistema para trabajar con flujos continuo de audio.
  - Desarrollo y aplicación de mejoras para la detección de locuciones cortas.
  - Desarrollo y aplicación de técnicas de adaptación de dominio de forma supervisada y no supervisada.
- Evaluación del rendimiento del sistema (y cada una de las adaptaciones/mejoras propuestas) mediante un conjunto de experimentos y protocolo de evaluación definido.
- Diseño y desarrollo de un demostrador gráfico.

Dicho sistema tendrá en cuenta la naturaleza específica del audio *broadcast* para optimizar el rendimiento en presencia de este tipo de audio.

Este sistema albergará técnicas utilizadas en el estado del arte actual para la tarea de reconocimiento de locutor, adaptadas a las características específicas para las que está diseñado debido a la fuerte restricción del entorno acústico y de grabación, así como locuciones muy cortas, solapamiento entre locutores, etc. Por tanto, se deberán explorar distintos métodos que permitan ajustar las técnicas de reconocimiento de locutor tradicionales y en el estado del arte, para implementar un sistema capaz de trabajar con dicha problemática.

Se tomará como punto de partida los sistemas de reconocimiento de locutor basados en i-vectors. Estos sistemas tienen un alto impacto en las tecnologías utilizadas actualmente y sostienen grandes resultados con entornos "controlados" en mayor o menor medida. Existen multitud de artículos que demuestran y motivan la necesidad de aplicar diferentes técnicas, tanto de pre-procesado, procesado a nivel de i-vectors, como de post-procesado de resultados. De cara a mejorar los resultados de los sistemas tradicionales en este tipo de entornos.

Por tanto, se propone una mejora en el rendimiento de este tipo de sistemas cuando las condiciones del audio tienden a no estar controladas, las transiciones entre locutores no estar claramente definidas, etc. Proporcionando un sistema base y unos resultados validos para un posible uso de este tipo de sistemas en situaciones en tiempo real.

Su desarrollo, además, involucra la articulación de los conocimientos, habilidades y destrezas adquiridos a lo largo de la formación del máster con asignaturas tales como Tecnologías del Habla, Reconocimiento Biométrico, Procesados Avanzando de Señales Multimedia entre otras. Adicionalmente aborda problemas propios del área de procesamiento de señales de voz y audio incorporando componentes de I+D+i. El trabajo involucra la realización de estudios, valoraciones e informes acerca de las tecnologías disponibles, innovaciones y alternativas.

1.3. Metodología y plan de trabajo

Con el objetivo de seguir una planificación estratégica que permita la consecución de los objetivos descritos anteriormente (véase sección 1.1) se propone la siguiente metodología y plan de trabajo:

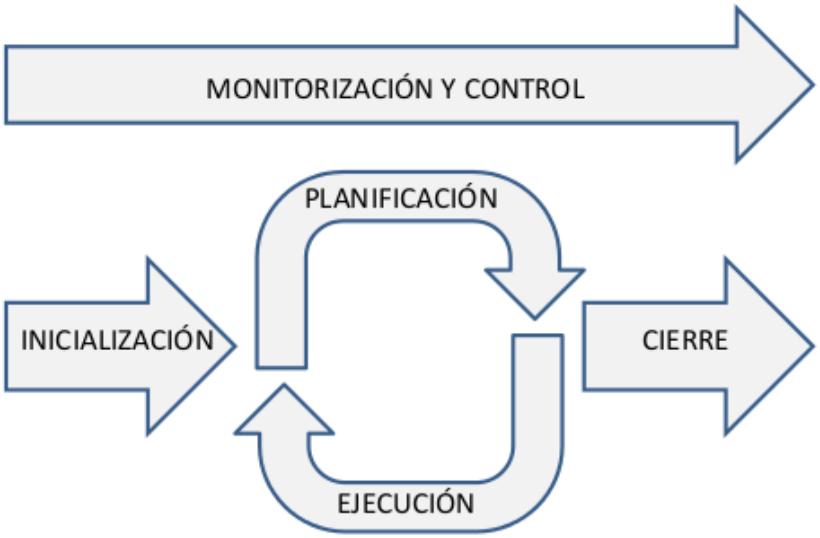

Figura 1.2: Diagrama de metodología y plan de trabajo seguido durante el proyecto.

- **Iniciación - 70 horas (del 25/09/2017 al 22/10/2017)**:
  - Propuesta del proyecto, identificación y análisis de la problemática a cubrir.
  - Estudio de tecnologías y técnicas presentes en el estado del arte que permitan implementar soluciones tecnológicas apropiadas.
  - Estudio del entorno de desarrollo y software. Principalmente lectura de documentación, instalación y puesta en un funcionamiento de sistema Linux, intérprete Bash, software Kaldi y Matlab.
- **Planificación - 20 horas (del 23/10/2017 al 25/12/2017)**:
  - Adquisición y descripción de las bases de datos a utilizar: Audias-Radio-2015, NIST SRE y Switchboard.
  - Estudio e interpretación de los datos, para la selección de locutores válidos en el sistema.
  - Diseño del sistema y selección de la configuración inicial.
- **Ejecución - 80 horas (del 01/11/2017 al 08/01/2018)**:
  - Implementación del sistema de detección de locutores en entorno Kaldi.
  - Adaptación del sistema para trabajar con flujos continuos de audio (programas).
  - Desarrollo y aplicación de mejoras para la detección de locuciones cortas en entorno Matlab.
  - Desarrollo y aplicación de técnicas de adaptación de dominio de forma supervisada y no supervisada en entorno Matlab.

- **Monitorización y control - 75 horas (del 25/09/2017 al 13/02/2018)**:
  - Control de errores y comprobación funcionalidades.
  - Pruebas objetivas del rendimiento del sistema para sus distintas aproximaciones.
  - Ajuste de parámetros, configuración e incorporación de nuevas funcionalidades.
  - Generación de informes de control y seguimiento.
- **Cierre - 55 horas (del 08/01/2018 al 13/02/2018)**:
  - Evaluación de resultados finales.
  - Recopilación de documentación, imágenes y datos.
  - Redacción de documento/memoria trabajo fin de máster.
  - Documentación extra, manual de uso y cierre del proyecto.

1.4. Organización de la memoria

El proyecto sigue la siguiente estructura y desglose por capítulos:

- Capítulo 1, Introducción: Introduce un breve resumen sobre la tarea que pretende llevar a cabo este proyecto, como la motivación y los objetivos del mismo.
- Capítulo 2, Sistemas de reconocimiento de locutor: Presenta de forma generalista los sistema de reconocimiento de locutor, como funcionan, cuales son sus aplicaciones, problemas y limitaciones. Así como sus tipos y modos de operación.
- Capítulo 3, Tecnología aplicada en sistemas de reconocimiento automático de locutor: Describe de forma técnica los diferentes bloques que representan un esquema de reconocimiento de automático utilizados en el estado del arte.
- Capítulo 4, Sistema diseño y desarrollo: Describe de la implementación de los sistemas diseñados, su parametrización y el propósito de cada una de las técnicas empleadas.
- Capítulo 5, Entorno experimental: Describe las bases de datos y las métricas de rendimiento que serán utilizadas para el análisis objetivo del rendimiento de los sistemas.
- Capítulo 6, Experimentos y resultados: Presenta los resultados obtenidos para cada experimento realizado y analiza las posibles mejoras o defectos en cada una de las aproximaciones propuestas.
- Capítulo 7, Conclusiones y trabajo futuro: Expone los principales resultados obtenidos y muestra los objetivos alcanzados durante el proyecto, incluyendo una propuesta de trabajo futuro.

Capítulo 2

Sistemas de reconocimiento de locutor

En este capítulo se presenta una introducción sobre los sistemas de reconocimiento de locutor. Para ello, se mostrará la estructura general de este tipo de sistemas, cuál es el objetivo principal de los mismos y su problemática asociada.

2.1. Introducción

Las tecnologías basadas en reconocimiento de locutor han ganado visibilidad e importancia para gran parte de la sociedad durante los últimos años. Su alto grado de aceptabilidad por los usuarios y el buen rendimiento que aplican este tipo de sistemas, los hacen óptimos para ser usados en multitud de aplicaciones.

En particular, la fácil adquisición de los datos y la baja intrusividad en su captación. Unido al desarrollo de tecnologías emergentes (smartphones, IoT, etc.) han permitido un creciente uso de servicios y aplicaciones que proporcionan sistemas personalizados por locutor (Apple Siri, Amazon Alexa, Google Home), control y seguridad, gestión de grandes cantidades de datos, así como reconocimiento forense.

Estos sistemas de reconocimiento de locutor tratan de emular el proceso reconocimiento de locutor humano, para proporcionar ciertos servicios al usuario con alta calidad. Por ello los sistemas automáticos presentan fortalezas y debilidades similares a los sistemas humanos que tratan de ser aprovechadas para mejorar la precisión en el reconocimiento.

Los seres humanos, habitualmente, son capaces de reconocer a personas por sus voces con sorprendente precisión, especialmente cuando el grado de familiaridad es alto (es decir, conocidos o figuras públicas), incluso en ocasiones es suficiente una expresión no lingüística, como una risa, para poder identificar a un locutor conocido. Por otro lado, cabe destacar la dificultad que implica reconocer una voz que se escuchó apenas un par de veces o una voz familiar por teléfono.

Esto corresponde a una habilidad cognitiva inherente que permite realizar una identificación efectiva y precisa de voces humanas que permite llevar a cabo una comunicación natural de persona a persona. Por ejemplo, al hablar por teléfono (no existe referencia visual) generalmente se comienza identificando quién está hablando y además se realiza una verificación subjetiva de que la identidad es correcta, permitiendo que la conversación pueda continuar.

Es por esto que existe un área de investigación dentro de la comunidad científica dedicada al procesado de señal de voz y específicamente al reconocimiento de locutor, que trata de implementar y mejorar sistemas de reconocimiento automático de locutor.

ASPECT	HUMANS	MACHINES
TRAINING	SPEAKER RECOGNITION IS AN ACQUIRED HUMAN TRAIT AND REQUIRES TRAINING.	REQUIRES SUFFICIENT DATA TO TRAIN THE RECOGNIZERS.
VAD	DIFFERENT PARTS OF THE HUMAN BRAIN ARE ACTIVATED WHEN SPEECH AND NONSPEECH STIMULI ARE PRESENTED.	SPEECH SIGNAL PROPERTIES AND STATISTICAL MODELS ARE USED TO DETECT PRESENCE OR ABSENCE OF SPEECH.
AUDIO PROCESSING	THE HUMAN BRAIN PERFORMS BOTH SPECTRAL AND TEMPORAL PROCESSING. IT IS NOT KNOWN EXACTLY HOW THE AUDIO SIGNAL DEVELOPS THE SPEAKER- OR PHONEME-DEPENDENT ABSTRACT REPRESENTATIONS/MODELS.	ACOUSTIC FEATURE PARAMETERS DEPENDING ON SPECTRAL AND TEMPORAL PROPERTIES OF THE AUDIO SIGNAL ARE UTILIZED FOR RECOGNITION.
HIGH-LEVEL FEATURES	WE CONSIDER LEXICON, INTONATION, PROSODY, AGE, GENDER, DIALECT, SPEAKING RATE, AND MANY OTHER PARALINGUISTIC ASPECTS OF SPEECH TO REMEMBER A PERSON'S VOICE.	RECENT ALGORITHMS HAVE INCORPORATED PROSODY, PRONUNCIATION, DIALECT, AND OTHER HIGH-LEVEL FEATURES FOR SPEAKER IDENTIFICATION.
COMPACT REPRESENTATION	THE HUMAN BRAIN FORMS SPEAKER-DEPENDENT, EFFICIENT ABSTRACT REPRESENTATIONS. THESE ARE INVARIANT TO CHANGES OF THE ACOUSTIC INPUT, PROVIDING ROBUSTNESS TO NOISE AND SIGNAL DISTORTION.	HIGH-LEVEL FEATURES ARE EXTRACTED THAT SUMMARIZE THE VOICE CHARACTERISTICS OF A SUBJECT. THESE ARE EXTRACTED IN A WAY TO MINIMIZE SESSION VARIABILITY DUE TO NOISE OR DISTORTION.
LANGUAGE DEPENDENCE	SPEAKER RECOGNITION BY HUMANS IS BETTER IF THEY KNOW THE LANGUAGE BEING SPOKEN.	AUTOMATIC SYSTEM'S PERFORMANCE IS DEGRADED IF THERE IS A MISMATCH IN TRAINING AND TEST LANGUAGE.
FAMILIAR VERSUS UNFAMILIAR SPEAKERS	HUMANS ARE EXTREMELY GOOD AT IDENTIFYING FAMILIAR VOICES, BUT NOT SO FOR UNFAMILIAR ONES.	MACHINES PROVIDE CONSISTENT PERFORMANCE WHEN ADEQUATE AMOUNT OF DATA IS PROVIDED. FAMILIARITY CAN BE RELATED TO THE AMOUNT OF TRAINING DATA.
IDENTIFICATION VERSUS DISCRIMINATION	THE HUMAN BRAIN PROCESSES THESE TWO TASKS DIFFERENTLY.	IN MOST CASES, THE SAME ALGORITHM (WITH SLIGHT MODIFICATION) CAN BE USED TO IDENTIFY AND DISCRIMINATE BETWEEN SPEAKERS.
MEMORY RETENTION	HUMANS' ABILITY TO REMEMBER A PERSON'S VOICE DEGRADES WITH TIME.	A COMPUTER ALGORITHM CAN STORE THE MODELS OF A PERSON INDEFINITELY IF PROVIDED SUPPORT.
FATIGUE	HUMAN LISTENERS CANNOT PERFORM AT THE SAME LEVEL FOR A LONG DURATION.	COMPUTERS DO NOT HAVE ISSUES WITH FATIGUE. LONG RUNTIMES MAY CAUSE OVERHEATING IF NECESSARY PRECAUTIONS ARE NOT TAKEN.
IDENTIFY IDIOSYNCRASIES	HUMANS ARE VERY GOOD AT IDENTIFYING CHARACTERISTIC TRAITS OF A VOICE.	THE MACHINE ALGORITHMS HAVE TO BE SPECIFICALLY TOLD WHAT TO LOOK FOR AND COMPARE.
MISMATCHED CONDITIONS	HUMANS RELY MORE ON PARALINGUISTIC ASPECTS OF SPEECH IN SEVERE MISMATCHED CONDITIONS.	AUTOMATIC SYSTEMS ARE TRAINED ON VARIOUS ACOUSTIC CONDITIONS, AND USUALLY ARE MORE ROBUST.
SUSCEPTIBILITY TO BIAS	HUMAN JUDGMENT CAN BE BIASED BY CONTEXTUAL INFORMATION.	AUTOMATIC ALGORITHMS CAN BE BIASED TOWARD THE TRAINING DATA.

Figura 2.1: Comparativa en el proceso de reconocimiento de locutor entre humano y máquina (extraído de [Hansen and Hasan, 2015]).

2.2. Tipos de sistemas

Los sistemas de reconocimiento de locutor tienen como objetivo determinar la pertenencia (o no) de una locución a un individuo conocido. Generalmente, siguen una estructura claramente definida. En primer lugar, se realiza una primera fase de entrenamiento que tiene como objetivo dar de alta a los usuarios en el sistema de reconocimiento (identidades de reconocer) y una fase de verificación (usuario corresponde con identidad reclamada) o identificación (usuario corresponde con alguna identidad).

Para ello, durante la fase de entrenamiento se realiza la extracción de características de la señal de voz que permitan discriminar unos locutores de otros y se generan los modelos (patrones) de los diferentes locutores que posteriormente deberán ser reconocidos por el sistema, ver Figura 2.2. Este entrenamiento puede realizarse con distintas alternativas, como se verá en el Capítulo 3.

Tras el registro de las identidades en el sistema, se procede a la fase de reconocimiento de locutor. Esta fase puede hacer uso de dos modos de operación distintos: verificación e identificación.

Los sistemas de verificación tratan de determinar si un locutor desconocido corresponde con un locutor específico (identidad reclamada). Por ello se denomina one-to-one mapping, pues realiza una comparación uno a uno y toma una decisión binaria (adaptado o rechazado), ver Figura 2.3.

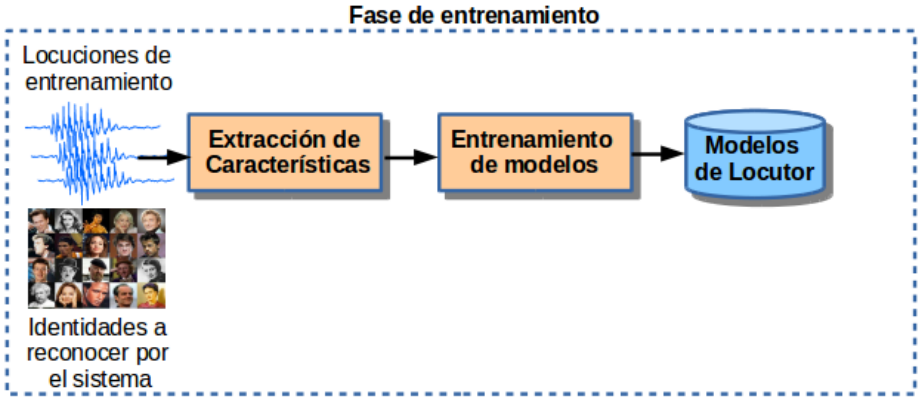

Figura 2.2: Esquema de funcionamiento de un sistema de reconocimiento de locutor clásico durante la fase de entrenamiento.

Estos sistemas, normalmente, se etiquetan como sistemas de conjunto abierto (*open-set*) ya que permiten diferenciar entre un locutor conocido y cualquier otro locutor desconocido.

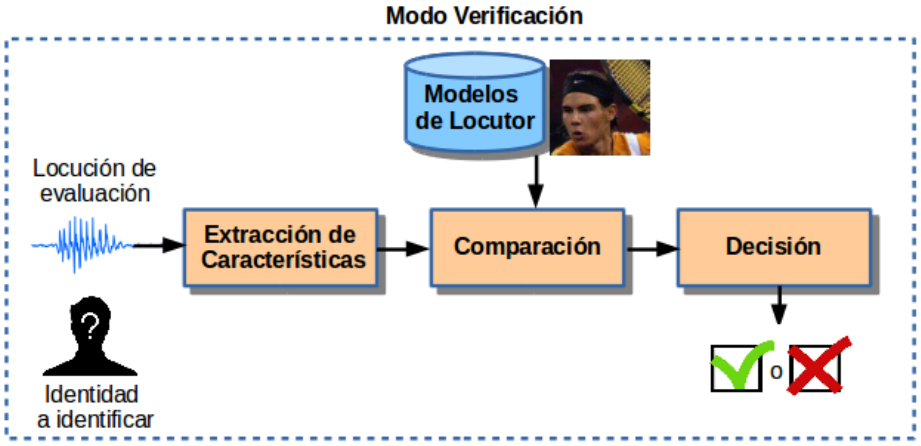

Figura 2.3: Esquema de funcionamiento de un sistema de reconocimiento de locutor clásico en modo verificación.

Los sistemas de identificación trabajan con el propósito de determinar si un locutor desconocido corresponde con alguno de los locutores conocidos por el sistema. Por ello, se denomina one-to-many mapping, pues realiza una comparación frente a multitud de locutores conocidos.

Por último, se tomará una decisión respecto a la identificación del locutor, es decir, si la similitud es suficientemente alta frente a alguno de los modelos, se podrá asociar esa locución al individuo representado por el modelo (patrón), ver Figura 2.4

Cuando la decisión del sistema siempre identifica alguno de los locutores conocidos se tratará de un sistema en conjunto cerrado (*closed-set*). En cambio, si el sistema es capaz de identificar locutores conocidos y además, si la similitud no es suficientemente alta descarta la identificación. Entonces se trata de un sistema en conjunto abierto (*open-set*) [Bimbot et al., 2004].

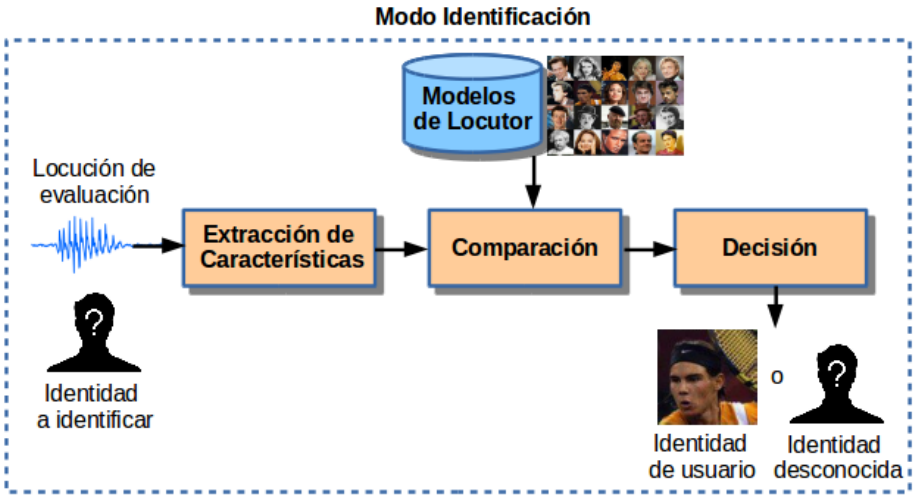

Figura 2.4: Esquema de funcionamiento de un sistema de reconocimiento de locutor clásico en modo identificación *open-set*.

2.2.1. Sistema detección de locutores

En particular, durante este proyecto se desarrollará un sistema de identificación de locutor en conjunto abierto (*open-set*) independiente de texto. Con la particularidad de trabajar con flujos continuos de audio y multitud de locutores desconocidos. Esto implica una connotación diferente a la subyacente tras los modos de identificación y verificación.

Este tipo de sistema de detección tiene como objetivo la identificación de cualquiera de los locutores dados de alta en el sistema. Donde esta la identificación se realiza sobre pequeñas ventanas temporales, permitiendo localizar de manera precisa las apariciones de cada uno de los locutores en cada instante temporal, ver Figura 2.5.

En contraposición frente a sistemas clásicos de reconocimiento de locutor, este tipo de sistemas de detección no trabajan con locuciones individuales sino que permiten realizar un análisis temporal de los locutores que aparecen en un flujo continuo de audio.

Por tanto, implican una cierta mejora significativa frente a otros esquemas de identificación. Dando la posibilidad de ubicar apariciones de locutores de manera precisa dentro de gran cantidad de audio.

Debido a la necesidad de realizar un análisis exhaustivo en pequeñas ventanas temporales, es necesario realizar una gran cantidad de comparaciones entres segmentos y modelos de locutor hipótesis (*trials*). Esto implica la necesidad de un mayor número de recursos computacionales frente a esquemas clásicos en virtud del número de locutores dados de alta en el sistema.

2.3. Aplicación de los sistemas de reconocimiento de locutor

En la actualidad existen multitud de ámbitos de aplicación para todo tipo de tecnologías de procesado de voz. En particular el reconocimiento de locutor abarca un gran área, tanto en aplicaciones reales ya utilizables, como en aplicaciones potencialmente desarrollables.

El abanico de posibilidades que ofrece de este tipo de sistemas permite el desarrollo de

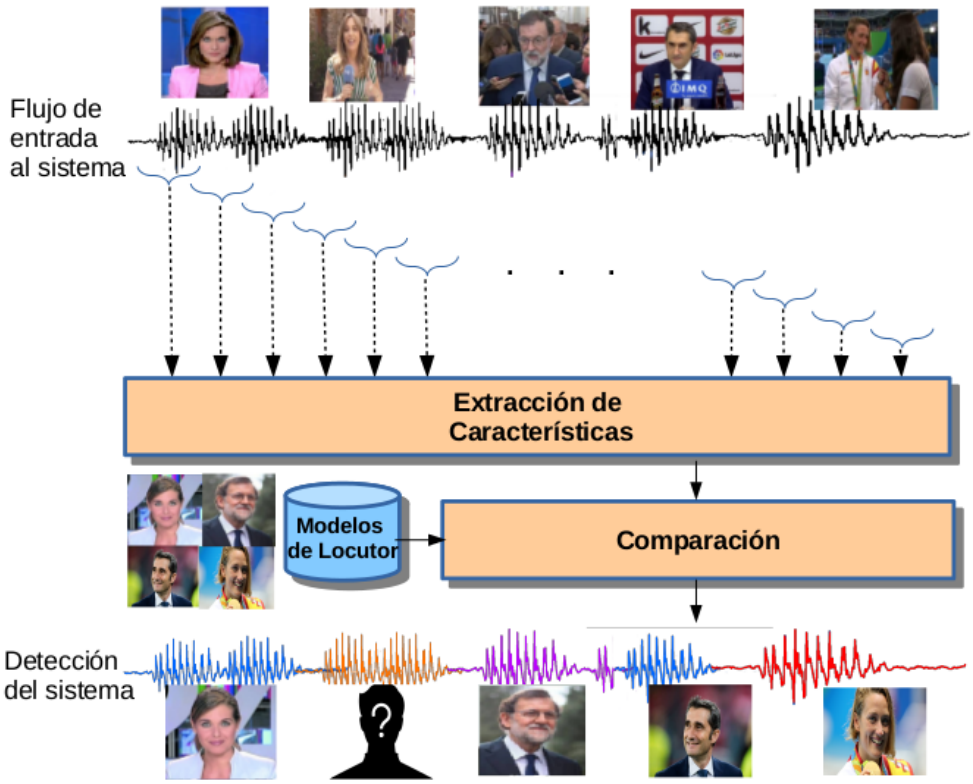

Figura 2.5: Esquema de sistema de detección de hablantes en audio *broadcast*.

cualquier aplicación susceptible hacer uso de un sistema de reconocimiento locutor. Por ejemplo alguno de los grandes campos donde se utilizan este tipo de sistemas son, [Reynolds, 2002]:

- Ámbito legal (ciencias forenses, libertad condicional en el hogar, etc.)
- Personalización (asistente personal, recomendaciones, etc.)
- Seguridad (control de acceso, sistemas de verificación, autenticación de transacciones bancarias, etc.)
- Gestión de datos de voz (generación automática de metadatos, búsqueda en bases de datos, etc.)

En particular, el sistema que se propone durante este proyecto principalmente se encuentra enmarcado sobre la tarea de gestión de datos de voz. Esto permite realizar consultas sobre bases de datos a partir de contenido, es decir, se podrán realizar búsquedas de un locutor específico presente en uno o varios flujos de audios (o programas radiofónicos). Por tanto, es posible obviar la necesidad de generar metadatos con los locutores que aparecen en uno o varios audios y además, posibilita la realización de búsquedas de manera rápida y eficiente de una identidad reclamada en un gran conjunto de datos. Permitiendo localizar de forma precisa las apariciones de cada uno de los locutores.

Además, otra de las ventajas de este tipo de sistemas basados en detección de locutores, se basa en la posibilidad de conocer que individuo aparece en cada uno de los instantes temporales de un audio de manera continua. Esto viene motivado por la necesidad inherente al

ser humano de validar la identidad del individuo con el que comparte un proceso comunicativo. Habitualmente esto ocurre cuando uno de los usuarios de una comunicación no es capaz de identificar la voz de otro de los usuarios y además no existe una referencia visual.

Sin embargo, esto también ocurre cuando se comparte una referencia visual del otro individuo, pero no se está familiarizado con su identidad. Un ejemplo claro de esto puede ser una programa de radio o televisión; durante un programa de radio se producen multitud de ocasiones donde se desconoce la identidad del interlocutor y sería beneficioso para el acto comunicativo una identificación de su identidad cuando no se tiene referencia visual. Por otro lado, durante un programa de televisión puede ocurrir lo mismo; entra en escena un interlocutor del que se tiene una referencia visual, pero no se puede obtener una verificación de su identidad por parte del espectador (poco familiarizado con el interlocutor).

Todos estos tipos de aplicaciones pueden basarse en sistemas de reconocimiento de locutor o incluir mejoras a partir de ellos.

Por tanto, puede definirse como una tecnología orientada a multitud de usos, en multitud de ámbitos y en creciente desarrollo. Esto se puede observar en empresas punteras como Google y Amazon, que utilizan estas tecnologías en la implementación de sus productos (Google Home, Amazon Alexa).

2.4. Problemas y limitaciones de los sistemas de reconocimiento de locutor

Es cierto que los sistemas de reconocimiento de locutor contemplan un alto grado de desempeño, aceptabilidad y fiabilidad. Pero existen restricciones y limitaciones en la usabilidad de estos sistemas debido a los siguientes factores:

- Variabilidad del entorno: distintos dispositivos de grabación (micrófonos), calidad del canal de transmisión (cable, GSM, satélite, etc.), ruido ambiente (SNR, distorsión).
- Cantidad de datos disponibles: número de locuciones, duración del audio, cantidad de locutores disponibles, etc.
- Duración de las locuciones: mayor duración, y variabilidad de las muestras de audio presentes en un sistema, tienden a mejorar su rendimiento.
- Cooperación de los locutores: Existen locutores que cambian su forma de hablar, provocan solapamiento con otros locutores e incluso imitan o falsean voces.
- Estabilidad a largo plazo: estados de ánimo, estado de salud (catarro, afonía, etc.)

Estos problemas y limitaciones deben tenerse en cuenta durante el desarrollo de los sistemas, sobre todo cuando se plantean con el objetivo de obtener una utilidad práctica. De tal forma, que se debe realizar un estudio previo de la tarea específica a realizar. Es necesario el desarrollo y verificado de un sistema apto para la tarea concreta a realizar.

Dichos sistemas de detección de hablantes en audio *broadcast* deben adaptarse a condiciones de alta variabilidad en cuanto a calidad de la señal de voz (distintos micrófonos, ruido ambiente, canal de transmisión, etc.) y segmentos de voz de corta duración (declaraciones, entrevistas, etc). Además, de obtener un rendimiento óptimo acorde a la cantidad de datos disponibles para su implementación. Por tanto, se deberá tener en cuenta esta problemática durante la ejecución del proyecto.

2.4.1. Entorno de alta variabilidad

Una de las dificultades que encuentran los sistemas de reconocimiento de locutor reside en la alta variabilidad en multitud de escenarios. Generalmente, el rendimiento de los sistemas suele ser mejor siempre y cuando estén diseñados para aplicaciones determinadas, por tanto, no es posible realizar un sistema de reconocimiento de locutor que permita mantener alta fiabilidad para todo tipo de entornos acústicos, de grabación, etc. Es por ello, que la alta variabilidad que se produce en audio de medios, es uno de los conceptos clave a tener en cuenta durante el desarrollo de estos sistemas. En particular un sistema de detección de locutores en audio *broadcast* recibe alta variabilidad de multitud de ámbitos:

- Variabilidad de canal: Los diferentes dispositivos de grabación (micrófono de estudio, teléfono móvil, grabadora, etc) producen variabilidad asociada a los canales de grabación y las técnicas asociadas a su codificación (submuestreo, GSM, satélite, etc).
- Variabilidad acústica: Las diferentes situaciones que en el entorno de grabación de audio de medios (solapamiento entre locutores, música de fondo, entrevistas en exterior, mítines, pasillos de congreso, ruedas de prensa, etc) aumentan la variabilidad asociada con la que debe trabajar el sistema.

Por todo esto, el sistema implementado debe ser suficientemente robusto frente a la variabilidad generada por fuentes externas y en diferentes entornos. Dotando al sistema de gran capacidad para minimizar la variabilidad intra-clase y aumentar la inter-clase, con el objetivo de maximizar la separación entre la representación de locutores distintos.

2.4.2. Locuciones cortas

Los sistemas de reconocimiento de locutor tienden a aumentar su rendimiento en relación con la longitud de las duraciones con las que trabajan. Es por ello, que los sistemas de detección de audio *broadcast* deben ser capaces de mantener unos resultados aceptables con duración de locuciones de relativamente cortas (1-10 segundos).

Esto se debe a la propia naturaleza del audio proveniente de medios de comunicación donde las locuciones tienden ser cortas (véase Tabla 5.2.3) y alternativamente entre distintos locutores. Esto implica situaciones donde se produce solapamiento entre locutores e interjecciones de duración mínima (menos de 1 segundo).

Este problemática complica la estimación de los modelos y la extracción de los vectores que representarán los segmentos de audio (i-vectors, véase Capitulo 3). Por tanto, es necesario desarrollar los sistemas de detección de forma que permitan identificar locuciones cortas, por ejemplo, a partir de segmentos cortos (de duración 5 segundos) que contengan gran cantidad de información proveniente de una misma locución. Además, resulta interesante tomar dichos segmentos con cierto solapamiento, esto es: los segmentos deben mantener información redundante compartida con sus predecesores y sucesivos segmentos. El objetivo de ellos es aumentar la robustez del sistema en el proceso de detección, (ver Figura 2.5).

2.4.3. Cantidad de datos disponibles en el dominio

Una de las partes fundamentales de los sistemas de reconocimiento de locutor es la cantidad de datos disponibles. Estos datos permiten construir sistemas robustos y fiables siempre y cuando la cantidad de datos sea suficientemente alta (miles de horas de audio, cientos de

locutores y decenas de locuciones por locutor). Para ello es necesario disponer de gran cantidad de audio etiquetado (referencia de locutores disponible) y no etiquetado (referencia de locutores no disponible).

Además, y de forma conveniente, los datos utilizados como soporte para el desarrollo del sistema deben provenir del mismo dominio (entorno, calidad, idioma, etc.) que los datos de evaluación (*test*) sobre los que se aplicará el sistema. De esta forma, los sistemas obtendrán mejores resultados de manera dependiente a la cantidad de datos disponibles y en relación con el ajuste de los datos utilizados en el desarrollo del mismo, frente al entorno de aplicación.

En particular, el hecho de disponer de gran cantidad de datos en el dominio de la aplicación es una tarea arduamente difícil que conlleva gran cantidad de coste en recursos. Por ello, se trabaja en técnicas que permitan mantener en el rendimiento de los sistemas a partir de su desarrollo con gran cantidad de datos fuera de dominio (*out-domain*) y utilizar una menor cantidad de datos del propio dominio (*in-domain*) con el objetivo de realizar una adaptación a la aplicación específica (*domain adaptation*), [Garcia-Romero and Espy-Wilson, 2011].

Alguna de estas técnicas se presentarán y utilizarán en el sistema desarrollado, véase Capitulo 4. Debido a la gran cantidad de datos disponibles para la realización de este proyecto en el dominio de habla telefónica en idioma inglés (*out-domain*) y la poca cantidad de datos en el dominio de audio *broadcast* en castellano (*in-domain*), véase Figura 1.1.

Capítulo 3

Tecnología aplicada en sistemas de reconocimiento automático de locutor

En esta sección se mostrarán las principales técnicas, tecnologías y algoritmos que contribuyen de manera directa en el desarrollo de los sistemas de reconocimiento de locutor utilizados recientemente en el estado del arte. Como tal, estas técnicas se encuentran directamente relacionadas con el sistema desarrollado en este proyecto y por tanto, serán descritas en detalle.

3.1. Estado del arte

La evolución de los sistemas de reconocimiento automático de locutor ha sido significativa a lo largo de los últimos años. A principios de los años 2000s, los sistemas de reconocimiento de locutor se basaban principalmente en modelos paramétricos puramente estadísticos extraídos a partir del analisis a corto plazo de señales espectrales (short-term spectral features). Estos sistemas utilizaban la relación de verosimilitudes como detector entre dos modelos GMM. Un modelo GMM adaptado a locutor frente a un modelo genérico GMM (denominado UBM) [Reynolds et al., 2000]. Hasta la llegada de nuevas técnicas basadas en Factor Analisys [Kenny and Dumouchel, 2004][Dehak et al., 2011], que supusieron una excelente mejora, tanto en resultados, como en rendimiento computacional.

3.1.1. Esquema general de sistemas i-vector

Los sistemas i-vectors principalmente tienen su fundamento en modelos generativos tipo GMM-UBM y técnicas de compensación de variabilidad sobre un subespacio de dimensionalidad reducida.

Típicamente, se presenta un bloque de extracción de características a partir de la señal de voz, seguido de una estructura GMM-UBM que permite realizar un modelado del comportamiento genérico para cualquier locutor, mediante pesos, medias y desviaciones típicas (π, μ, Σ, respectivamente). Tras ello, se produce la extracción de i-vectors (una representación de una locución dentro de un subespacio de variabilidad total) a partir de la denominada matriz de *total variability* (T), entrenada con gran cantidad de locuciones y los estadísticos del GMM-UBM. Una vez realizada la extracción de i-vectors es posible aplicar normalización de media (m) y/ó whitenning (W) [Garcia-Romero and Espy-Wilson, 2011]. Por último, se realiza el proceso de puntuación (*scoring*) que permite calcular el grado de similitud entre dos i-vectors dados (entrenamiento frente a evaluación). En esta etapa es posible aplicar distintas

métricas como: *cosine scoring*, LDA (*Linear Discriminant Analysis*) ó PLDA (*Probabilistic Linear Discriminant Analisys*), esta última realiza una compensación de variabilidad a partir de matrices de covarianza inter-clase (*across-class, AC*) e intra-clase (*within-class, WC*) entrenadas previamente de forma supervisada, ver Figura 3.1.

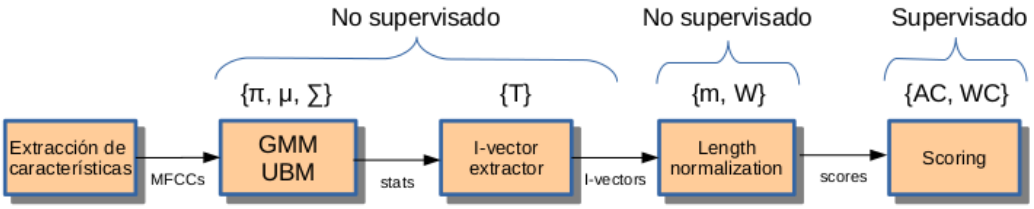

Figura 3.1: Diagrama de bloques de sistema de reconocimiento de locutor basado en i-vectors. Representación de sus hiper-parámetros y tipo de entrenamiento (supervisado/no supervisado).

3.1.2. Extracción de características

Los sistemas de reconocimiento automático de locutor toman como entrada la forma de onda que se emite durante el proceso de producción de voz humano, que contiene multitud de características de forma intrínseca. Habitualmente los modelos estadísticos como GMMs, no son capaces de manejar grandes cantidades de caraterísticas, es decir, no serían capaces de modelar correctamente con datos que provengan directamente de la forma de onda emitida por un locutor. Además, se produce un problema denominado "maldición de la dimensionalidad", que indica un crecimiento exponencial del número de muestras de entrenamiento necesarias según el número de características utilizadas [Campbell, 1997] [Kinnunen and Li, 2010].

Es por ello necesario realizar una extracción de características que permitan realizar una discriminación de locutores de la forma más óptima posible, por tanto, las características ideales deberían cumplir:

- Alta variabilidad inter-locutor y baja variabilidad intra-locutor.
- Robustez frente a ruido y distorsión.
- Frecuencia en el proceso de habla espontánea.
- Fácil extracción a partir de la señal de voz.
- Difícilmente reproducible por un individuo impostor.
- Invariante a cambios en la voz producidos por cuestiones de salud, edad, etc.

Desde un punto de vista de interpretación física, se pueden dividir cinco categorías de características:

- Características espectrales localizadas a corto plazo (*short-term spectral features*).
- Características de la fuente de voz.
- Características espectro-temporales

- Características prosódicas.
- Características de alto nivel.

En el estado del arte, habitualmente, se seleccionan características espectrales localizadas a corto plazo (LPCCs, MFCCs principalmente), debido a su fácil cómputo y buen rendimiento. La desventaja de estas características es su baja robustez frente al ruido y distorsión, que se tratará de solventar en etapas posteriores de compensación de variabilidad [Jain et al., 2000][Reynolds, 2003].

Linear Prediction Cepstral Coefficients - LPCCs

El análisis LPC (*Linear Prediction Coefficients*) se basa en el modelado del proceso de producción de voz humano mediante la estimación de la envolvente espectral de la señal de voz. Dicha envolvente puede representarse como un filtro autorregresivo (AR) todo-polos [Fant, 1970]. Por tanto, el análisis LP consiste en estimar los coeficientes de dicho filtro en cada ventana de análisis, formando un vector de características localizado.

Para ello, se aplica un filtro pre-énfasis a la señal de voz, que permite realzar altas frecuencias del espectro con el objetivo que mitigar el efecto contratio que se produce en el proceso de producción de voz [Rabiner and Juang, 1993].

$$x_p(t) = x(t) - a \cdot x(t-1) \tag{3.1}$$

donde a típicamente se encuentra en el intervalo [0.95, 0.98].

Tras esto, se lleva a cabo un enventanado de la señal en tramas (20 a 30 milisegundos), normalmente con solapamiento entre ellas (50 %), donde la señal se considera pseudo-estacionaria para su análisis en frecuencia. Cada trama es multiplicada por una función de enventanado (tipo Hamming), con el objetivo de atenuar la señal en sus extremos para realizar el correcto análisis en frecuencia.

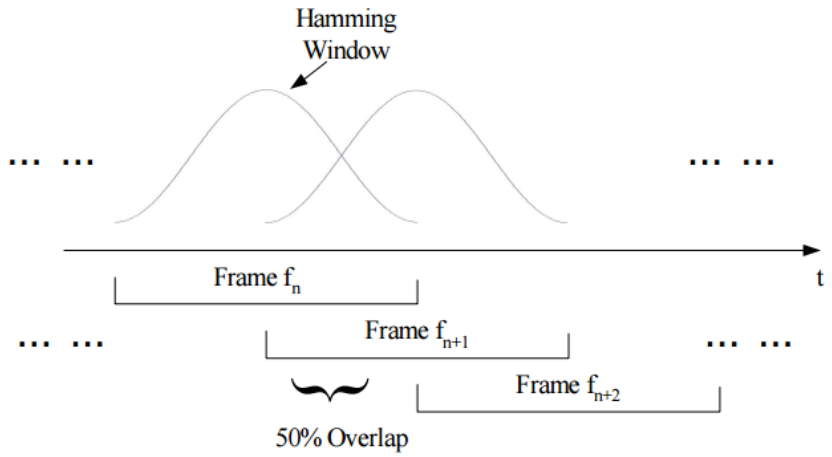

Figura 3.2: Proceso de enventanado de la señal de voz con ventana Hamming y solapamiento 50 %, para la extracción de características cepstrales LPCC y MFCC, (extraído de [Bimbot et al., 2004]).

Teniendo como entrada cada una de las tramas previamente enventanadas, se aplica análisis LP mediante recursión Levinson-Durbin para resolver las ecuaciones que surgen de la

formulación de mínimos cuadrados. Este cálculo de los coeficientes de predicción lineales (LPC) a menudo se denomina método de autocorrelación.

A partir de estos coeficientes LPC es posible calcular su transformación cepstral y por tanto los LPCC, como:

$$c_0 = \ln \sigma^2 \tag{3.2}$$

$$c_m = a_m + \Sigma_{k=1}^{m-1} \binom{k}{m} c_k a_{m-k}, \quad 1 \leq m \leq p \tag{3.3}$$

$$c_m = \Sigma_{k=1}^{m-1} \binom{k}{m} c_k a_{m-k}, \quad p < m \tag{3.4}$$

donde σ^2 es el término de ganancia del modelo LPC, a_m son los coeficientes LPC y p es el número de coeficientes LPC calculados.

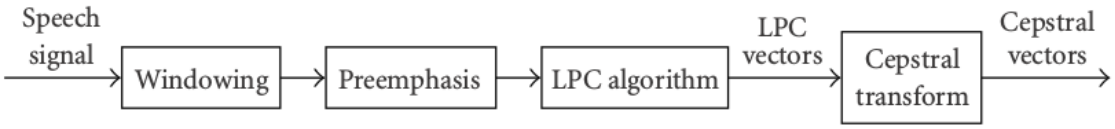

Figura 3.3: Esquema de extracción de características LPCC (extraído de [Bimbot et al., 2004]).

Mel Frequency Cepstral Coefficients - MFCCs

La extracción de características MFCC (*Mel Frequency Cepstral Coefficients*) se basa en la estimación de la envolvente espectral mediante el cálculo de la transformada discreta de Fourier y la aplicación de un banco de filtros perceptutal en escala Mel.

Para ello, (de igual forma que LPCC) se aplica un filtro pre-énfasis a la señal de voz que permite realzar altas frecuencias del espectro con el objetivo que mitigar el efecto contrario que se produce en el proceso de producción de voz, véase ecuación 3.1.

Tras esto, se lleva a cabo un enventanado de la señal en tramas (20 a 30 milisegundos), normalmente con solapamiento entre ellas (50 %), donde la señal se considera pseudo-estacionaria para su análisis en frecuencia. Cada trama es multiplicada por una función de enventanado (Hamming), con el objetivo de atenuar la señal en sus extremos para realizar el correcto análisis en frecuencia.

A partir de estas tramas de voz se realiza la estimación de la envolvente espectral en base a la Transformada Rápida de Fourier (FFT, por sus siglas en inglés)[Schaefer and Oppenheim, 1989], de la cual se extrae únicamente su módulo. Acto seguido, para reducir el tamaño del vector espectral y realizar un suavizado del mismo, se aplica un banco de filtros en escala Mel (filtros triangulares equiespaciados en una escala de frecuencias con base perceptual auditiva humana):

$$f_{MEL} = 1000 \cdot \frac{log(1 + \frac{f_{lineal}}{1000})}{log2} \tag{3.5}$$

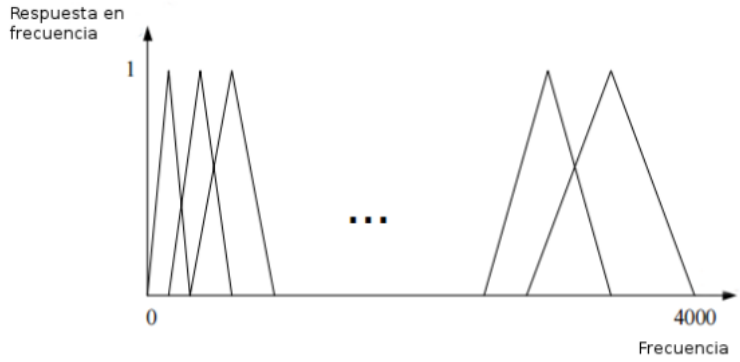

Figura 3.4: Banco de filtros en escala Mel sobre escala natural.

Donde a cuya salida, se obtiene la envolvente espectral en decibelios (dB) tomando 20*log 10 de la salida de cada filtro. Quedando representada de forma compacta la envolvente espectral por cada ventana de análisis en los coeficientes MFCC.

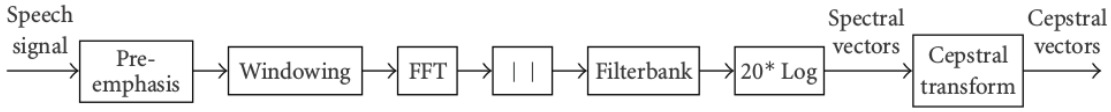

Figura 3.5: Esquema de extracción de características MFCCs , (extraído de [Bimbot et al., 2004]).

Por último, en ambos casos (LPCC y MFCC) se aplica la Transformada Discreta del Coseno (DCT, por sus siglas en inglés), según:

$$c_n = \sum_{k=1}^{K} S_k \cdot cos[n(k - \frac{1}{2})\frac{\pi}{K}], n = 1, 2, ..., L \tag{3.6}$$

Esta expresión compacta la información presente en los K coeficientes MFCC S_k o LPC, respectivamente, en un número menor de valores ($L \leq K$) por trama. Además, es habitual aplicar una normalización de media y varianza (*Cepstral Mean and Variance Normalization, CMVN*) con el objetivo de mitigar la componente aditiva del canal de transmisión:

$$c'_n = \frac{c_n - \mu_n}{\sigma_n} \tag{3.7}$$

Donde μ_n corresponde con la media y σ_n con la desviación típica y c'_n el coeficiente n normalizado.

Información dinámica (Δ, ΔΔ)

Una vez calculados los coeficientes cepstrales y aplicado CMVN, es de relevancia añadir cierta información dinámica. Esta información indica como varían los vectores de características en el tiempo. Habitualmente se realiza mediante una aproximación de la primera (velocidad) y segunda derivada (aceleración)[Furui, 1981], Δ y ΔΔ respectivamente:

$$\Delta c_m = \frac{\sum_{k=-l}^{l} k \cdot c_{m+k}}{\sum_{k=-l}^{l} |k|} \tag{3.8}$$

$$\Delta\Delta c_m = \frac{\sum_{k=-l}^{l} k^2 \cdot c_{m+k}}{\sum_{k=-l}^{l} k^2} \tag{3.9}$$

Estas características se añaden a las previamente calculadas en el análisis MFCC o LPCC y sirven como vector de características de entrada al sistema.

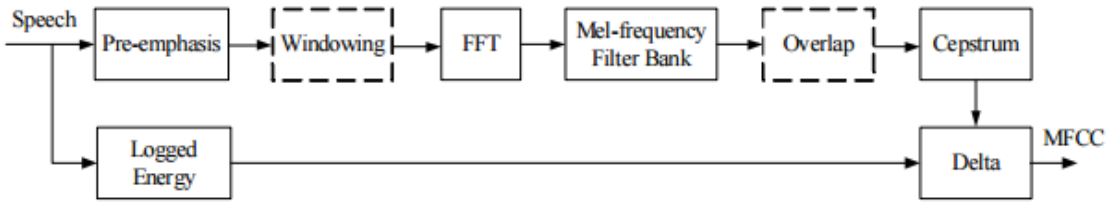

Figura 3.6: Esquema de extracción de características MFCCs y adición de coeficientes Δ, $\Delta\Delta$, , (extraído de [Bimbot et al., 2004]).

3.1.3. Estructura GMM-UBM

Con el objetivo de dotar al sistema de información suficiente para crear un background de conocimiento sólido acerca del comportamiento genérico de las características a modelar, se proponen los modelos generativos basados en GMM-UBM (*Gaussian Mixture Models - Universal Background Model*). Este proceso necesita gran cantidad de datos y suele conllevar un tiempo de computación alto. Pero es indispensable aportar este tipo de información al sistema, que permitirá construir un espacio de alta dimensionalidad como base suficientemente sólida en el modelado de la tarea a realizar.

Gaussian Mixture Models - GMMs

Una forma de representar la distribución de características de un modelo de locutor determinado es mediante un modelo GMM (*Gaussian Mixture Model*). Estos modelos probabilísticos generativos estocásticos permiten representar densidades arbitrarias para densidades multivariables. Asumiendo una distribución que viene dada por mezclas de Gaussianas, permiten construir un espacio de características representativo de los datos. El proceso de entrenamiento de estos modelos consiste en la estimación de sus parámetros: peso π, media μ y varianza Σ, que definen una función densidad de probabilidad dada por:

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \tag{3.10}$$

La función densidad de probabilidad es una combinación lineal de K densidades Gaussianas, $\mathcal{N}(x|\mu_k, \Sigma_k)$, ponderadas por pesos, π_k (también denominados probabilidades *a priori*). Cada una de estas Gaussianas viene parametrizada por un vector media $D \times 1$, μ_k y una matriz de covarianza $D \times D$, Σ_k:

$$\mathcal{N}(x|\mu_k, \Sigma_k) = \frac{1}{(2\pi)^{\frac{D}{2}}|\Sigma_k|^{\frac{1}{2}}} exp\{-\frac{1}{2}(x-\mu_k)'(\Sigma_k)-1(x-\mu_k)\} \tag{3.11}$$

donde los pesos de las mezclas satisfacen la restricción $\sum_{k=1}^{K} \pi_k = 1$.

Por motivos de rendimiento computacional y beneficiándose de la ortogonalidad presente en los coeficientes cepstrales MFCC (independencia entre dimensiones), se permiten representar las matrices de covarianza Σ de dimensión $D \times D$ como matrices diagonales. Esto fuerza una estimación de las Gaussianas con curvas de contorno circulares en 2-dimensiones (esféricas en 3-dimensiones), produciendo un mínimo error en posteriores etapas.

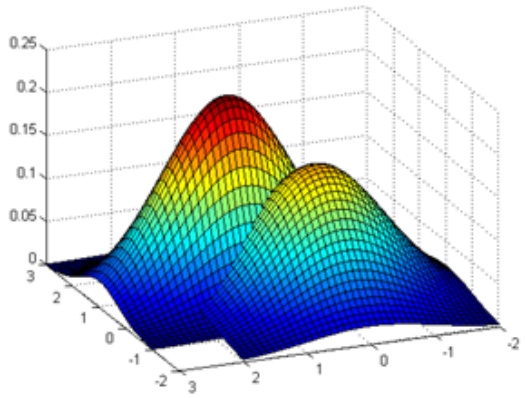

Figura 3.7: Ejemplo de GMM con mezcla de 2 Gaussianas en espacio de 3-dimensiones.

El proceso de entrenamiento de un modelo GMM habitualmente se apoya en el algoritmo *Expectation-Maximization* (EM). Este algoritmo realiza la estimación de los parámetros de la mezcla de Gaussianas dado un número fijo de componentas Gaussianas a estimar y una gran cantidad de datos. EM es un algoritmo iterativo que realiza dos etapas [Moon, 1996][Dempster et al., 1977]:

- E-step: Computo de las probabilidades *a posteriori*, como:

$$h_m^{(j)}(t) = \frac{w_m^{(j)}\mathcal{N}(x^{(t)}; \mu_m^{(j)}, \Sigma_m^j)}{\sum_{i=1}^{N} w_i^{(j)}\mathcal{N}(x^{(t)}; \mu_i^{(j)}, \Sigma_i^j)} \tag{3.12}$$

- M-step: Reestimación de los parámetros del modelo, $\lambda = \{\pi_k, \mu_k, \Sigma_k\}$, como:

$$w_k^{(j+1)} = \frac{1}{N}\sum_{t=1}^{N} h_k^{(j)}(t) \tag{3.13}$$

$$\mu_k^{(j+1)} = \frac{\sum_{t=1}^{N} h_k^{(j)}(t)x^{(t)}}{\sum_{t=1}^{N} h_k^{(j)}(t)} \tag{3.14}$$

$$\Sigma_k^{(j+1)} = \frac{\sum_{t=1}^{N} h_k^{(j)}(t)[x^{(t)} - \mu_k^{(j)}][x^{(t)} - \mu_k^{(j)}]^T}{\sum_{t=1}^{N} h_k^{(j)}(t)} \tag{3.15}$$

donde j indica el número de iteración, x^t el dato para $t = 1, ..., N$, siendo N el número de muestras y k determina la componente Gaussiana.

Habitualmente, con el objetivo de reducir el número de iteraciones necesarias hasta alcanzar la convergencia, se inicializa mediante *k-means* [Hartigan and Wong, 1979]. Esto permite una estimación de partida sobre los parámetros λ. Para ello, haciendo uso del número de Gaussianas del modelo (K), se realiza un agrupamiento (*clustering*) de los datos, donde se asocia cada uno de ellos al centroide más cercano. Estos centroides representarán la primera estimación de medias, $\mu_k^{(0)}$. Las matrices de covarianza se extraen de las covarianzas de los datos asociados a cada centroide $\Sigma_k^{(0)}$ y los pesos según el número de datos en cada agrupación (*cluster*), $w_k^{(0)}$.

Universal Background Model - UBM

El *Universal Background Model*, UBM, se define como un gran GMM entrenado para representar la distribución de características de un modelo universal de locutor. Por tanto, se busca seleccionar suficiente cantidad de voz que permita realizar un modelado donde se concentre gran cantidad de variabilidad proveniente de multitud de locutores y condiciones acústicas [Reynolds et al., 2000]. Es decir, se construye un GMM de forma que sea capaz de representar un número indefinido de locutores buscando localizar los parámetros μ_k, Σ_k dentro del espacio D-dimensional del GMM que permita modelar el comportamiento genérico de cualquier locutor.

Para realizar el entrenamiento UBM se busca estimar los parámetros $\lambda = \{\pi_k, \mu_k, \Sigma_k)$ mediante un proceso iterativo *Expectation-Maximization* (EM), ver ecuaciones 3.12-3.15, que permite refinar los parámetros de manera iterativa para aumentar monótonamente la similitud del modelo estimado y así converger rápidamente.

Con esto se consigue obtener un modelo genérico y robusto, que podrá ser adaptado con menor cantidad de datos para cada locutor específico.

Maximimum a posteriori - MAP adaptation

Habitualmente la cantidad de datos disponibles para generar un modelo de locutor válido vía GMM es insuficiente. Por este motivo se hace uso de técnicas como MAP (*Maximimum A Posteriori*) que permiten adaptar un modelo genérico entrenado con gran cantidad de datos (UBM) para estimar un modelo específico por locutor. Para ello es necesario adaptar los parámetros del modelo, $\lambda = \{\pi_k, \mu_k, \Sigma_k)$, aunque en la práctica habitualmente se realiza únicamente una adaptación de medias (μ_k), debido a un compromiso entre eficiencia computacional y rendimiento. A partir de un modelo UBM y unos datos de entrenamiento, se realiza la adaptación por cada locutor de la siguiente forma:

$$\mu_k^{adapt} = \alpha_k \frac{1}{n_k} f_k + (1 - \alpha_k)\mu_k \tag{3.16}$$

siendo,

$$\alpha_k = \frac{n_k}{n_k + \tau} \tag{3.17}$$

$$n_k = \sum_{t=1}^{T} P_{kt} \tag{3.18}$$

$$f_k = \sum_{t=1}^{T} P_{kt} x_t \tag{3.19}$$

$$P_{kt} = \frac{w_k p_k(X_t)}{\sum_{k=1}^{K} p_k(x_t)} \tag{3.20}$$

donde τ indica el factor MAP de adaptación, P_{kt} la probabilidad de ocupación de la Gaussiana y n_k y f_k se definen como los estadísticos de primer y segundo orden, respectivamente.

Esto permite un desplazamiento de μ_k a μ_k^{adapt} en el espacio D-dimensional del UBM, ver Figura 3.8. Disponiendo un modelo por locutor mucho más robusto, que el que se obtendría con un entrenamiento de GMM con pocos datos.

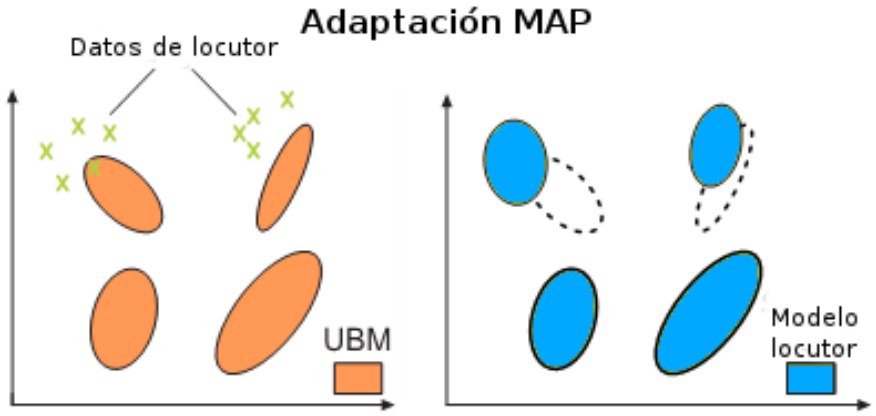

Figura 3.8: Ejemplo de UBM-MAP. Adaptación a un modelo UBM a partir de datos de entrenamiento de un locutor dado, (extraído de [Hansen and Hasan, 2015]).

Supervectores

Dado un esquema GMM-UBM(MAP) que realiza una adaptación de medias, es posible realizar la representación de un modelo GMM únicamente haciendo uso de las mismas. Esto implica generar un vector de medias en un espacio de alta dimensión ($D_{Gaussianas} \times N_{DimensionDatos}$), denominado supervector:

$$supervector_\lambda = \begin{bmatrix} \mu_1 \\ \mu_2 \\ . \\ . \\ . \\ \mu_n \end{bmatrix} \tag{3.21}$$

Dicho supervector se forma a partir de la concatenación de los vectores de medias, con dimensión igual al número de características ($D \times 1$). Estos vectores de medias provienen de K Gaussianas presentes en los modelos GMM, por tanto, se obtiene un supervector de $K \cdot D$ dimensiones. Permitiendo concentrar toda la información de un GMM en un único vector de alta dimensión (ej.: en el sistema desarrollado en este proyecto 122880×1, con K=2048 y D=60).

3.1.4. Compensación de variabilidad

Factor Analysis - FA

Uno de los grandes problemas presentes en los sistemas de reconocimiento de locutor basados en características espectrales (*short-term spectral features*) se debe a la variabilidad producida por las diferentes condiciones (canal de transmisión, micrófonos, etc.). La técnica *Factor Analysis* (FA) trata de caracterizar la variabilidad como una variable contínua que implica poder modelar variabilidad inter e intra-clase de diferente manera [Kenny et al., 2007] [Kenny, 2005]. Por tanto, si FA recoge una técnica de compensación de variabilidad en el dominio de los supervectores y se toma la suposición de que la variabilidad se encuentra en un subespacio de menor dimensión. Es posible modelar tanto la variabilidad inter-sesión e inter-locutor, de forma que:

$$\mu_{s,h} = \mu_s + Vy_s + Ux_{s,h} \tag{3.22}$$

donde, μ_s corresponde con el supervector entrenado, μ_s con el supervector de locutor, V es el subespacio de variabilidad de locutor, U es el subespacio de variabilidad de sesión, x e y corresponde con los factores de sesión y ruido respectivamente.

Esto implica cierta limitaciones como, que el conjunto de entrenamiento deba ser representativo de la variabilidad real, así como un manejo intratable de técnicas clásicas de proyección en subespacios (LDA, etc.). Incluso que la estimación de la matriz U (variabilidad de canal) pueda contener información de locutor.

Total variability Subspace

La solución al problema definido en FA pasa por realizar una reducción de dimensionalidad a un subespacio de variabilidad total (*total variability subespace*), TV, que permite modelar de forma conjunta canal y locutor:

$$\mu_{locutor} = \mu_{UBM} + Tw \tag{3.23}$$

donde $m_{locutor}$ es el modelo total de la locución, m_{UBM} representa el UBM, T es la matriz de variabilidad total y w es el vector que más aproxima a $m_{locutor}$, denominado i-vector (por ser una representación en dimensionalidad intermedia entre supervectores y MFCCs).

3.1.5. I-vectors

Los i-vectors (abreviatura de vector de identidad en inglés)[Dehak et al., 2011] permiten realizar la representación de una locución en un único vector de dimensionalidad reducida y fija (típicamente 400 o 600 dimensiones). Estos vectores contienen la información de variabilidad de sesión, como de locutor de manera conjunta. Por tanto, se puede decir que contiene los factores o variables latentes del modelo.

La principales ventajas del uso de i-vectors residen en su reducida dimensionalidad (400-600 dimensiones) frente a los supervectores (aprox. 120000 dimensiones) y en una estimación del subespacio de locutor y sesión de manera conjunta frente a FA (subespacios de locutor y sesión separados). Esto permite la utilización de técnicas clásicas de proyección como LDA (Linear Discriminant Analysis) y una menor cantidad de datos para su estimación. Por tanto, su rendimiento computacional es mucho más elevado que técnicas predecesoras.

Aun así, el proceso de entrenamiento del subespacio de variabilidad total (matriz T), necesita gran cantidad de datos y conlleva un alto coste computacional (aunque mejora técnicas predecesoras). Su principal ventaja es la extracción de i-vectors de entrenamiento y evaluación de manera sencilla una vez se dispone de la matriz T entrenada, y permite identificar cualquier locución (sea cual sea su duración) mediante un vector de dimensión fija. Esto permite aplicar reglas de puntuación entre dos i-vectors (dos locuciones) en un subespacio que contiene la variabilidad total.

Intuitivamente, la "calidad" del i-vector viene determinada por la duración de la locución. Es decir, locuciones de mayor duración implican versiones menos "ruidosas" en el espacio de los i-vectors.

Normalización de i-vectors

Con el objetivo de mejorar la estimación de los i-vectors se suele realizar una normalización en longitud a partir de la proyección de los i-vectors en una función de densidad simétrica y esférica.

Para ello se aplica un proceso denominado *Whitening*. Este proceso permite realizar una transformación de los datos con el objetivo de obtener una matriz de covarianza unidad, a partir los operadores m y W calculados sobre un conjunto de i-vectors de entrenamiento.

Donde m se define como la media global de todos los i-vectors de entrenamiento y W como la raíz cuadrada inversa de la matriz de covarianza global (Σ_{wh}) para los datos de entrenamiento:

$$m = \frac{w_k}{K} \tag{3.24}$$

donde m se corresponde con el i-vector medio de todo el conjunto de entrenamiento y w_k con cada uno de los i-vectors de entrenamiento, para $k = 1, ..., K$.

$$W = \frac{1}{\sqrt{D}} \cdot V \tag{3.25}$$

donde W se corresponde con la matriz que normaliza a longitud unidad. Con D autovalores de la matriz de covarianza global y V los autovectores de la matriz de covarianza global para los i-vectors de entrenamiento y que por tanto cumplen: $\Sigma_{wh} \cdot V = V \cdot D$.

Por tanto, este proceso se realiza de forma no supervisada y debe utilizar para su entrenamiento un conjunto de i-vectors seleccionados específicamente para el cálculo de sus parámetros.

3.1.6. Puntuación

Una vez extraídos os i-vectors a partir de datos de entrenamiento y test, y aplicada su respectiva normalización. Se debe realizar una comparación entre ellos que mostrará el grado de similitud entre ambos. Esta comparación tendrá como resultados una puntuación (*score*) por cada pareja de i-vector de entrenamiento frente a test (*trial*) y permitirá determinar si una locución pertenece al mismo locutor o no.

Cosine scoring

Una de las formas clásica de puntación de i-vectors es la conocida como *cosine scoring*, que permite medir la similitud entre i-vectors mediante distancia coseno. Dado que la longitud de los i-vectors es totalmente independiente de la duración de las locuciones, es posible establecer una medida de similitud entre dos locuciones, sean cual sea sus duraciones, a partir de sus i-vectors de la siguiente manera:

$$score = cos(w_{locutor}, w_{test}) = \frac{< w_{locutor}, w_{test} >}{\|w_{locutor}\| \, \|w_{test}\|} \tag{3.26}$$

Linear Discriminant Analysis - LDA

Esta técnica tiene como objetivo la reducción de dimensionalidad preservando la información discriminatoria entre clases (esta es la principal diferencia con PCA, *Principal Component Analysis*).

A partir de i-vectors de entrenamiento, se obtiene una matriz de proyección LDA que permite maximizar la separación entre clases dentro del subespacio de los i-vectors.

Esta matriz LDA permite realizar una proyección de los i-vectors de test:

$$w_{test}^{LDA} = S \cdot w_{test} \tag{3.27}$$

donde S es la matriz de proyección y w_{test}^{LDA} es una proyección del i-vector original w_{test}

Una vez aplicada la proyección se aplica *cosine scoring* de manera similar a ecuación 3.26:

$$score_{LDA} = cos(Sw_{locutor}, Sw_{test}) = \frac{< Sw_{locutor}, Sw_{test} >}{\|Sw_{locutor}\| \, \|Sw_{test}\|} = \frac{< Sw_{locutor}, w_{test}^{LDA} >}{\|Sw_{locutor}\| \, \left\|w_{test}^{LDA}\right\|} \tag{3.28}$$

Probabilistic Linear Discriminant Analysis - PLDA

Actualmente se trata de la técnica más utilizada en el estado del arte para la fase de puntuación. Tiene como objetivo el modelado de los subespacios de canal y locutor (inter-clase e intra-clase) en el espacio de variabilidad total (i-vectors).

Se basa principalmente en la técnica de Factor Analysis, FA (Sección 3.1.4), en este caso aplicado al subespacio de los i-vectors en lugar de al espacio de los supervectores.

$$w_{ij} = w + Fh_i + Gk_{ij} + E_{ij} \tag{3.29}$$

donde, w_{ij} corresponde con el i-vector entrenado, w con el i-vector media, F es el subespacio de variabilidad de locutor, G es el subespacio de variabilidad de sesión, k y h corresponde con los factores de sesión y locutor respectivamente y E corresponde a ruido.

Esta técnica sigue un proceso de entrenamiento para calcular las matrices F(Φ) (covarianza intra-clase) y G (Σ) (covarianza inter-clase) [Garcia-Romero and Espy-Wilson, 2011], utilizando datos de entrenamiento etiquetados por locutor. Estos datos permiten inicializar dichas matrices mediante PCA (previniendo un comportamiento determinista y evitando la necesidad de gran cantidad de estimaciones en el algoritmo EM para converger) y posteriormente se aplica el citado algoritmo EM (Expectation-Maximization).

Una vez obtenidas las matrices de covarianza F(Φ) (covarianza intra-clase) y G (Σ) (covarianza inter-clase) se calcula la puntuación entre dos i-vectors (entrenamiento vs evaluación) como:

$$score_{PLDA} = w_1^T Q w_1 + w_2^T Q w_2 + 2 w_1^T P w_2 + const \tag{3.30}$$

con

$$Q = \Sigma_{tot}^{-1} - (\Sigma_{tot} - \Sigma_{ac}\Sigma_{tot}^{-1}\Sigma_{ac})^{-1} \tag{3.31}$$

$$P = \Sigma_{tot}^{-1}\Sigma_{ac}(\Sigma_{tot} - \Sigma_{ac}\Sigma_{tot}^{-1}\Sigma_{ac})^{-1}, \tag{3.32}$$

donde $\Sigma_{tot} = \Phi\Phi^T + \Sigma$ y $\Sigma_{tot} = \Phi\Phi^T$.

3.2. Tareas adicionales

3.2.1. Adaptación de Dominio

Los avances en los modelos basados en subespacios de variabilidad, particularmente los sistemas i-vector, han demostrado una mejora sustancial en el rendimiento de los sistema de reconocimiento de locutor. Lamentablemente, estas técnicas son altamente dependientes de la cantidad de datos disponibles en el proceso de entrenamiento.

Esto da lugar a la necesidad de disponer de cientos de locutores con decenas de locuciones por cada uno de ellos con el objetivo de entrenar de manera robusta los distintos hiperparámetros (UBM, T, PLDA , etc.). Sin embargo, es totalmente idealista esperar obtener un gran conjunto de datos etiquetados y en condiciones similares para un nuevo entorno donde se desea aplicar un sistema de reconocimiento de locutor a una nueva aplicación.

Con el objetivo de compensar la falta de grandes cantidades de datos en nuevos entornos, surge la tarea denominada "Adaptación de dominio" (*Domain Adaptation*). Dicha tarea trata de adaptar sistemas previamente entrenados para un dominio específico (ej: voz telefónica), a un nuevo dominio del que se dispone poca cantidad de datos (ej: audio *broadcast*).

Es por ello que actualmente se dedican esfuerzos en investigación sobre técnicas que permitan utilizar un pequeño conjunto de datos para adaptar sistemas de reconocimiento de locutor basados en i-vectors.

Estas líneas de investigación sufrieron un gran empuje a partir del workshop [Adaptation, 2013] dando lugar a multitud de publicaciones con técnicas aplicables a "Adaptación de Dominio", entre las que destacan:

- *PLDA parameter interpolation, Fully Bayesian adaptation, Approximate MAP adaptation, Weighted Likelihood* [Garcia-Romero and McCree, 2014] [Shum et al., 2014].
- *Dataset variability compensation* [Aronowitz, 2014]
- *Variatonal Bayes Methods* [Villalba and Lleida, 2014]

Todas estás técnicas muestran una clara mejora cuando la adaptación se realiza sobre la etapa de puntuación y compensación de variabilidad (tipicamente PLDA). Mostrando una recuperación de hasta un 85 % del rendimiento en sistemas adaptados (ante un sistema *full in-domain* completamente optimista)[Garcia-Romero et al., 2014].

El resto de hiper-parámetros tienen menor impacto en la adaptación de los sistemas, por tanto, habitualmente no suelen ser adaptados si no es para realizar un ajuste fino o utilizar gran cantidad de datos sin etiquetar.

En particular, la técnica empleada durante este proyecto se trata de interpolación de matrices de covarianza PLDA [Garcia-Romero and McCree, 2014]. En dicho estudio ([Garcia-Romero and McCree, 2014]), se presenta una mejora sustancial del rendimiento del sistema cuando se realiza adaptación de dominio a partir de esta técnica. En este caso, se utilizan dos bases de datos SRE, como *in-domain* y Switchboard, como *out-domain* (véase Capítulo 5).

Como principal propiedad destacable entre de ambas bases de datos se encuentra la similitud de dominio, además de una gran cantidad de datos disponibles para ambos dominio (aunque no se utilizan en su totalidad para las pruebas descritas). Es decir, ambos conjuntos de datos provienen de un dominio similar, como es voz telefónica conversacional en inglés, del que se dispone gran cantidad de datos. Esto permite construir un entorno de evaluación muy bien determinado durante todo el artículo [Garcia-Romero and McCree, 2014].

Por tanto, se deberá comprobar si las mejoras mostradas en este artículo son extrapolables al sistema propuesto durante este proyecto, donde ambos dominios son altamente disjuntos (voz telefónica en inglés frente habla microfónica en castellano).

Estos resultados demuestran una recuperación de entre el 45-90 % del rendimiento máximo del sistema (con un entrenamiento completo *in-domain*, utilizando la base de datos SRE completa), ver Figura 3.9. Y además, este nivel de adaptación permite ser variable según el valor α como parámetro de interpolación, ver Figura 3.10.

Por tanto, uno de los objetivos de este proyecto será validar este tipo de técnicas cuando los datos disponibles en los diferentes dominios provengan de entornos totalmente distintos. Así como, determinar si la cantidad de datos disponibles implica una mejora sustancial en el rendimiento de la adaptación.

Adicionalmente, con el objetivo de aumentar la cantidad de datos utilizables durante el proceso de aprendizaje supervisado, se propone hacer uso de técnicas de *clustering*. Estas técnicas permiten realizar un etiquetado "automático" de los datos con el objetivo de ser utilizables en tareas supervisadas. Por tanto, aplicando algoritmos como: *Agglomerative Hierarchical Clustering* (AHC), *Infomap* y *Markov Clustering (MCL)*, combinados con las técnicas aplicables a Adaptación de Dominio, se consiguen resultados cercanos al 85 % del rendimiento obtenido por el sistema cuando se disponen de las etiquetas reales [Shum et al., 2014].

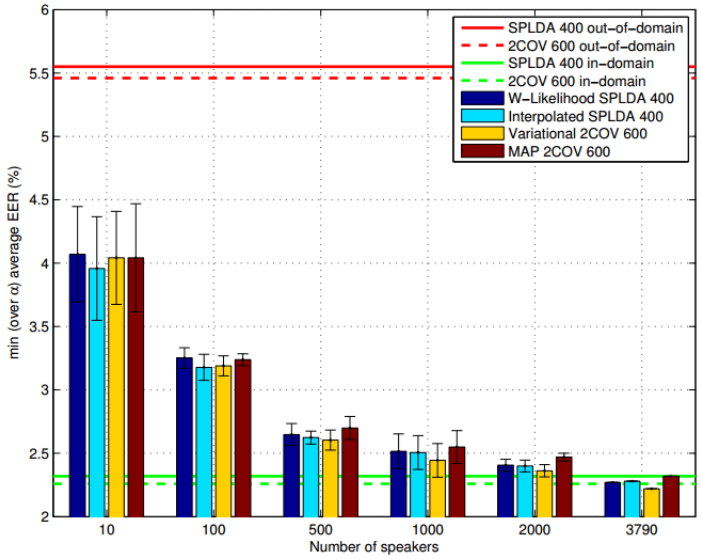

Figura 3.9: (Fuente:[Garcia-Romero and McCree, 2014]). Resultados comparativos de las distintas técnicas propuestas para adaptación de dominio. Rendimiento máximo *out-domain* (rojo) y rendimiento máximo *in-domain* (verde).

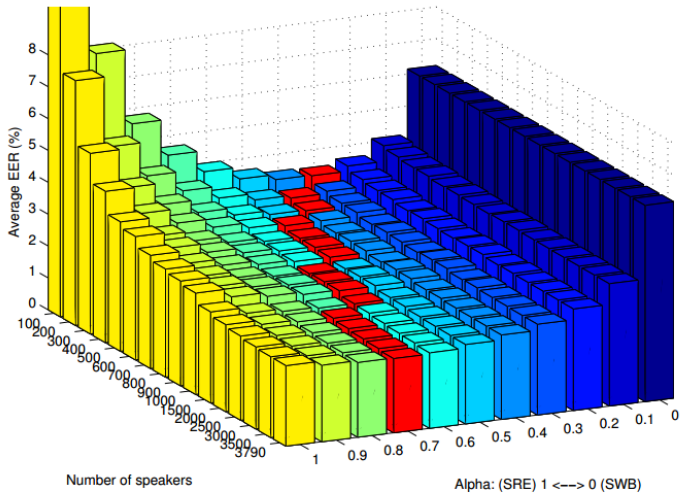

Figura 3.10: (Fuente:[Garcia-Romero and McCree, 2014]). Resultado de adaptación de dominio según α (parámetros de adaptación) en función del número de hablantes para SRE y SWB en [Garcia-Romero and McCree, 2014].

Capítulo 4

Sistema, diseño y desarrollo

Durante este capítulo se detallará el diseño y desarrollo de un sistema de detección de hablantes en locuciones cortas a través de tres aproximaciones: Sistema *Baseline*, que detalla la implementación de un sistema de reconocimiento automático basado en extracción de i-vectors. Sistema Locuciones Cortas, detallará modificaciones propuestas sobre el sistema *baseline* con el objetivo de trabajar con locuciones de corta duración. Sistema Adaptación de Dominio, mostrará las ideas claves implementadas para realizar la adaptación de un dominio genérico (voz telefónica) a un nuevo dominio (audio *broadcast*).

Para la realización de estos sistemas se aplicarán técnicas y procesos descritos en el Capítulo 3, que serán implementados en el entorno de desarrollo Kaldi [Povey et al., 2011], (véase sección 5.1), con ciertas funcionalidades en entorno Matlab^{TM}.

4.1. Sistema Baseline

Este sistema denominado Baseline, se diseña e implementa como un sistema genérico de reconocimiento de locutor. Es desarrollado como primera aproximación a un sistema de detección de hablantes en locuciones cortas en audio *broadcast* sin estar específicamente diseñado para ello. Utilizándose como sistema base para la adaptación a los sistemas propuestos en apartados posteriores.

Se basa en una estructura GMM-UBM que incorpora una estrategia de reducción de dimensionalidad TV (*Total Variability*) con el objetivo de modelar variabilidad de locutor y sesión de forma única. Obteniendo por tanto, una representación por cada locución en forma de i-vector.

Para ello, es posible definir cuatro fases claramente diferenciables, ver Figura 4.1.

- La primera fase de **desarrollo** o *background*, es la encargada de dotar robustez y base fundamental al sistema. Para ello, se realiza el entrenamiento de todos los hiperparámetros del sistema (UBM, T, m, W, AC, WC), utilizando gran cantidad de información proveniente de los **datos de desarrollo** (ver sección 5.3). En primer lugar, se realiza la extracción de características MFCC, y posteriormente se aplica un VAD (*Voice Activity Detector*) con el objetivo de descartar aquellas tramas que no sean susceptibles de contener voz. Estas características permiten realizar el entrenamiento del UBM.

  A partir de los estadísticos Baum-Welch extraídos del UBM y los MFCCs de desarrollo, se procede a realizar el entrenamiento de la matriz T (matriz de variabilidad total). Dicha matriz T permitirá realizar la extracción de los i-vectors.

Una vez se obtienen los i-vectors de los datos de desarrollo, se procede a hacer uso de ellos para obtener los hiper-parámetros de normalización (m y W) y realizar el entrenamiento del bloque PLDA, -matrices AC (*across-class*) y WC (*within-class*)-, que permitirán realizar la compensación de variabilidad durante la etapa de puntuación PLDA.

- La segunda fase, denominada **entrenamiento**, es la encargada de generar los i-vectors que representarán a cada uno de los locutores a detectar por el sistema. Para ello, tomará los **datos de entrenamiento** (locuciones de los locutores a detectar) (ver sección 5.3) y procederá a la extracción de características MFCC y procesado del VAD, de la misma manera que los datos de desarrollo.

  A partir de la matriz T generada en la fase de desarrollo, se procede a la extracción de los i-vectors y su posterior normalización mediante los parámetros m y W. Estos i-vectors de entrenamiento permiten obtener un único vector que representará el modelo de cada locutor, ver Figura 4.5 (izquierda).

- La tercera fase de **evaluación**, permite la extracción de i-vectors sobre las datos de evaluación. Estos **datos de evaluación** serán locuciones pre-segmentadas de locutores conocidos para el sistema *baseline* y posteriormente pasará a trabajarse con flujos continuos de audio como se detalla en la sección 4.2. Sea cual sea el audio de entrada, el procedimiento a realizar el similar. De igual forma que en la otras dos fases ya detalladas, se realiza la extracción de características MFCC y su posterior procesado VAD. A partir de la matriz T, se realiza la extracción de i-vectors y su normalización mediante m y W.

- La última fase de **puntuación** (*scoring*) permite obtener una medida de similitud (*score*) entre las diferentes muestras de entrenamiento y evaluación (*trials*). Para ello, toma cada una de los i-vectors provenientes de evaluación y realiza su comparación con cada uno de los modelos de locutor disponibles (i-vectors de entrenamiento).

  El sistema implementado utiliza tres bloques de puntuación completamente independientes, con el objetivo de realizar un estudio comparativo de los resultados que se obtienen con el uso de cada uno de ellos, -PLDA *scoring*, LDA *scoring* y *cosine scoring*-, ver sección 3.1.6.

  Esta medida de similitud (*scores*), podrá ser procesada de manera opcional por un bloque de normalización (z-norm), que permita alinear todos los *scores* en un mismo rango de valores. El objetivo de esto, es poder aplicar un umbral independiente de locutor para la decisión final del sistema (clasificación sobre a que locutor pertenece cada muestra de evaluación).

A continuación, se detallará de forma particular como se han implementado cada uno de los bloques presentes en estas etapas y que parámetros han sido utilizados para la realización de este proyecto.

4.1.1. Extracción de características

Se cuenta con un bloque de paramatrización MFCC-$\Delta, \Delta\Delta$, extractor de 60 coeficientes (20 MFCCs+20Δ+20$\Delta\Delta$) para cada trama de 20ms con solapamiento 10ms (50 %) y ventana Hamming. Adicionalmente se aplica una normalización de media y varianza (*Cepstral Mean and Variance Normalization*, CMVN) a cada coeficiente, con el objetivo de eliminar el efecto del canal de transmisión.

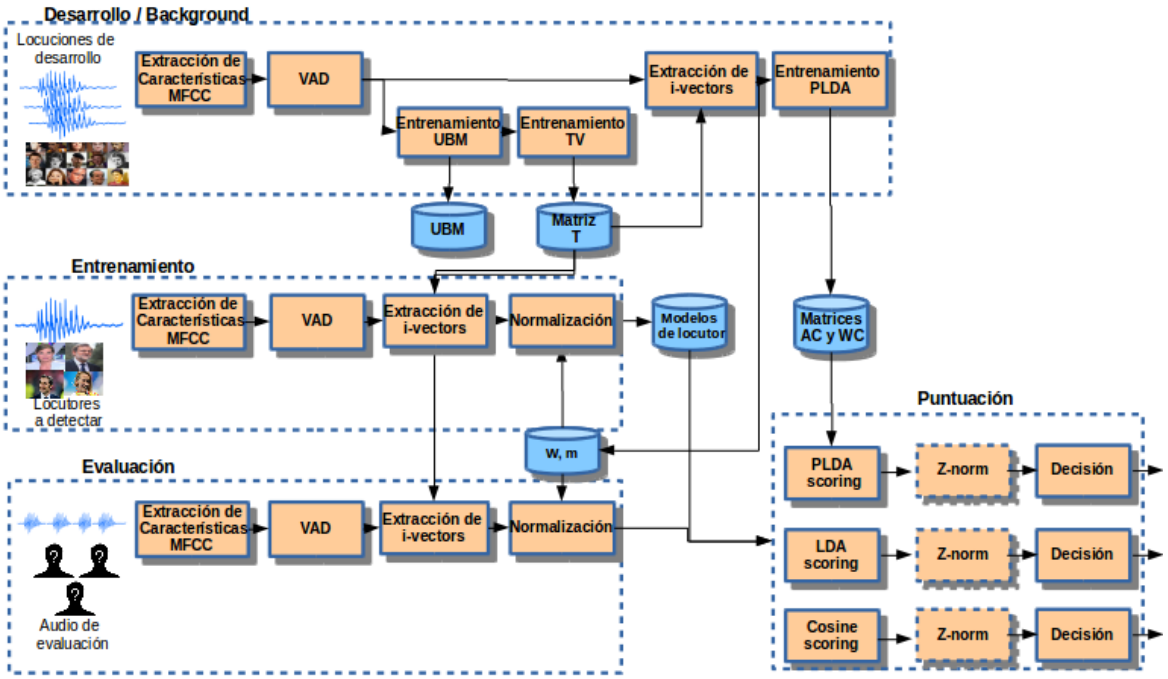

Figura 4.1: Esquema y estructura general del sistema de detección de hablantes desarrollado.

A su vez, se aplica un VAD (*Voice Activity Detector*), que permite descartar todas aquellas tramas que no sean susceptibles de contener voz y que por tanto, no contienen información relevante para el sistema.

4.1.2. *Universal Background Model* - UBM

Haciendo uso de las características extraídas sobre los datos de desarrollo disponibles, se procede al entrenamiento del UBM, compuesto por un GMM de 2048 componentes Gaussianas. Este UBM modela el comportamiento genérico de locutores dentro del dominio acústico correspondiente a los datos, en este caso voz telefónica.

En primer término, y para realizar el entrenamiento de forma más óptima, se realiza un primer paso de entrenamiento con matriz de covarianza diagonal (*diagonal-covariance UBM*) con 20 iteraciones EM. Posteriormente, a partir del diagonal-UBM, se entrena un modelo UBM con matriz de covarianza completa (*full-covariance UBM*) aplicando 4 iteraciones EM.

Este GMM-UBM completo permite una mejor representación de la características acústicas, pero es más costoso computacionalmente. Por tanto, se realizan menos iteraciones del algoritmo EM, lo que permite aumentar eficiencia computacional.

4.1.3. *Total Variability Subspace*

Para realizar posteriormente la extracción de i-vectors, se debe entrenar la matriz T (matriz de variabilidad total). Esta técnica de compensación de variabilidad, permite obtener un subespacio de dimensionalidad reducida.

Para ello se calculan los estadísticos Baum-Welch de 1er y 2o orden a partir del full-UBM calculado anteriormente y se deriva un subespacio TV de 600 dimensiones usando un análisis PCA y 5 iteraciones EM.

4.1.4. Extracción i-vectors

A partir de las características de entrada (MFCC) y la matriz T previa, se deriva la extracción de i-vectors para los subconjuntos de desarrollo, entrenamiento y evaluación.

La extracción de i-vectors nos permite mapear cada locución a su representación en un nuevo subespacio de variabilidad total.

Este subespacio modela de forma conjunta la variabilidad de canal y locutor y además permite trabajar en un subespacio de dimensionalidad reducida. Por tanto, cada locución de entrada quedará representada por un i-vector de dimensión fija, 600.

En el caso del subconjunto de entrenamiento, se calcula un i-vector medio por locutor w_{spk}, con todos los i-vectors correspondientes al mismo locutor.

$$w_{spk} = \frac{1}{n_s}\Sigma_{i=1}^{n_s}(w_i^s) \tag{4.1}$$

donde n_s corresponde al número de locuciones por locutor y s al número de locutores.

4.1.5. Scoring

Para determinar la similitud en dos i-vectors (entrenamiento vs evaluación), se ha optado por implementar las tres técnicas descritas en el Capitulo 3, *cosine scoring*, LDA y PLDA.

En concreto PLDA realizan una fase de entrenamiento supervisado (necesita datos etiquetados), para calcular las matrices de covarianza inter e intra-clase. En principio los datos corresponden con los datos de desarrolla, pero se realizarán pruebas con diferentes opciones como se especificará en cada una de las pruebas realizadas en el Capítulo 6.

Normalización cero (z-norm)

Con el objetivo de poder aplicar un umbral independiente de locutor en el proceso de clasificación. Se propone la técnica de normalización z-norm. Esta técnica tiene como objetivo corregir el posible desalineamiento de *scores* entre los distintos locutores. Por tanto, permite desplazar los scores de tal forma que estén contenidos en un rango de valores similar.

Para ello, para cada locutor y a partir de un conjunto de *scores* que provienen de *trials non-target*, se calculan los parámetros μ_{z-norm} y σ_{z-norm}, media y varianza de los *scores non-target*. Aplicando la siguiente ecuación a todos los *scores* de cada locutor, se obtiene una nueva distribución donde los *scores non-target* quedan centrados en cero con desviación típica unidad:

$$score_{z-norm} = \frac{score - \mu_{znorm}}{\sigma_{z-norm}} \tag{4.2}$$

Aplicando esta fórmula a todos los *scores* por locutor, se consiguen alinear todas las distribuciones impostor (*non-target*) del sistema.

4.2. Sistema Locuciones Cortas

El sistema de locuciones cortas partirá del esquema *baseline* descrito anteriormente, incorporando un enfoque orientado a detección de locutores en segmentos de corta duración.

Esto se debe a la propia naturaleza del audio *broadcast*, donde las locuciones tienden a ser breves y existen multitud de cambios de locutor en segmentos de audio relativamente cortos.

A partir de la Figura 5.2.3, se observa que los datos de evaluación provendrán de una distribución de duración por locuciones que tiende a estar centrada en segmentos de duración 1-10 segundos, con mediana en 5 segundos de duración por locución.

Teniendo en cuenta sistemas clásicos de reconocimiento de locutor que típicamente trabajan con duraciones superiores a 20 segundos para una correcta identificación de locutores. En este caso, es indispensable plantear un sistema de detección basado en extracción de i-vectors de corta duración. Para ello, es importante aumentar la resolución en el dominio de los i-vector con el objetivo de contrarrestar su posible mala estimación (i-vectors ruidosos) debida la corta duración de los mismos, mediante un esquema de solapamiento, ver Sección 4.2.1.

4.2.1. Extracción de i-vectors de corta duración con aumento de resolución para datos de evaluación.

Siguiendo las características en duración por locución de la base de datos Audias-Radio 2015, se propone una extracción de i-vectors con segmentos de duración 5 segundos (mediana de la distribución), reforzado con un aumento en la resolución mediante un solapamiento de 1 segundo entre ellos, véase Figura 4.2.

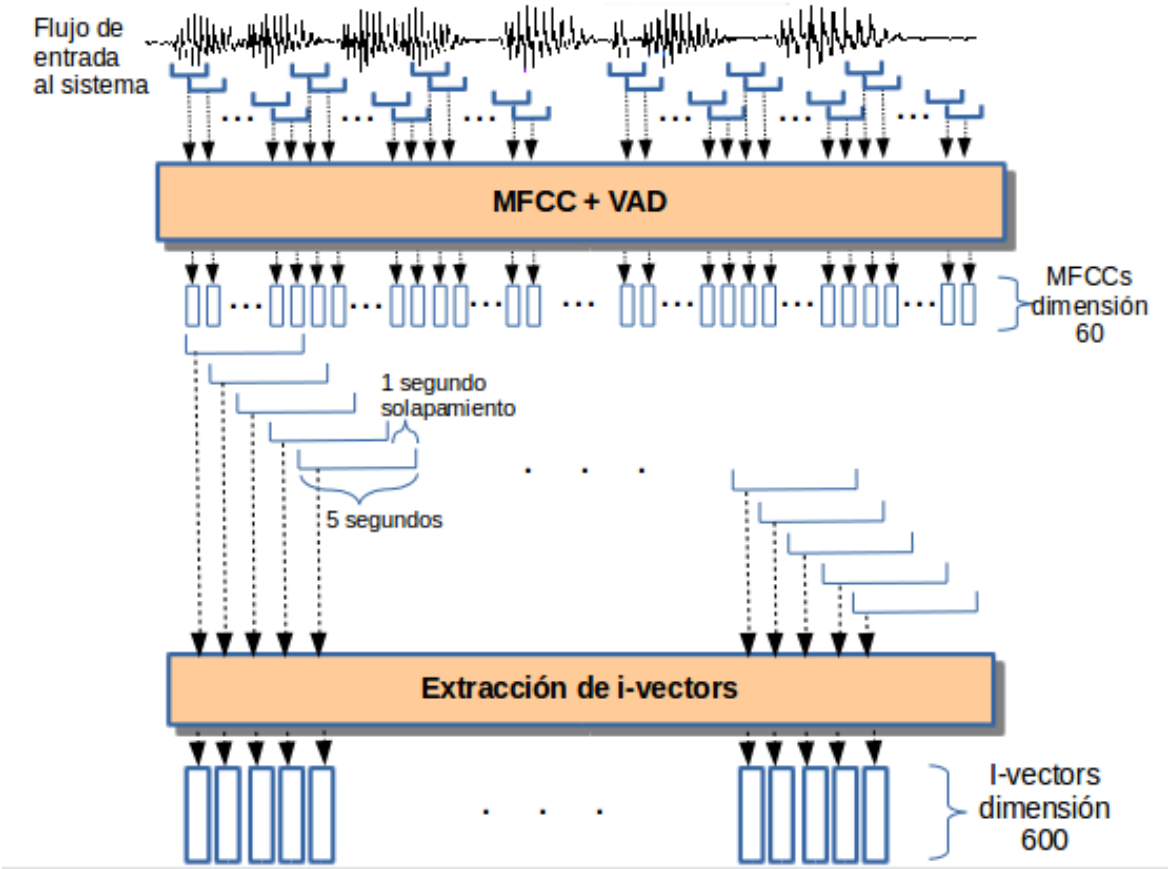

Figura 4.2: Esquema de extracción de i-vectors de corta duración (5 segundos) con aumento de resolución (solapamiento 1 segundo).

Por tanto, el sistema recibe un flujo continuo de audio, extrayendo una tasa de 1 ivector/segundo. Donde las estimaciones de cada i-vector se realizan a partir de un segmento de audio de 5 segundos, ver Figura 4.2. La idea subyacente trata de beneficiarse de la alta densidad en la extracción de i-vectors para contrarrestar la corta duración de las locuciones y los cambios frecuentes de locutor que pueden estar desalineados con la extracción de los i-vectors. Esto implica poder extraer información proveniente de 5 segundos de audio, con el beneficio obtener una alta resolución de 1 segundo.

4.2.2. Promediado de i-vectors de evaluación

Con el objetivo de optimizar el uso de la información presente en i-vectors solapados, se aplica un procedimiento de promediado de i-vectors.

Asumiendo un baja variación entre i-vectors consecutivos (diferencia de 1 segundos entre ellos), se supone una mejor estimación de cada segmento utilizando la información de i-vectors vecinos. Por tanto, el promediado de i-vectors pretende utilizar información contextual (i-vectors anteriores y posteriores) para mejorar su estimación.

Siguiendo la misma técnica aplicada en i-vectors de entrenamiento , donde se realiza un promediado de todos los i-vectors de entrenamiento con el objetivo de caracterizar mejor al locutor. En este caso, se realiza con el objetivo de incorporar información redundante de los i-vectors vecinos sobre un i-vector central.

Para ello, se localizan los N-ivectors en un vecindario dado y se sustituye el i-vector central por el promediado de todos los i-vectors vecinos, véase Figura 4.3.

Se aplican dos aproximaciones posibles: promediado lineal y promediado Hamming. Donde se hace uso de diferentes ponderaciones en función del tipo de promediado, de la siguiente forma:

$$w_{promedio}(n) = w_(n) \cdot v(n) \tag{4.3}$$

donde v se define como:

$$v_{rectangular}(n) = \begin{cases} 1, & -\frac{N-1}{2} < n < \frac{N-1}{2} \\ 0, & resto \end{cases}$$

para promediado lineal,

$$v_{hamming}(n) = 0{,}54 - 0{,}46\cos(2\pi\frac{n}{N}), \quad -\frac{N-1}{2} \leq n \leq \frac{N-1}{2} \tag{4.4}$$

para promediado Hamming.

4.2.3. Promediado de *scores*

A partir de un agrupamiento de *scores* se persigue realizar un suavizado sobre la salida del sistema, eliminando cierta variación entre *scores* muy dispares entre sí. Aplicando la misma idea vista anteriormente en el promediado de i-vectors, se busca utilizar información de contexto en el dominio de los *scores*, con el objetivo de eliminar variaciones rápidas entre detección/no detección.

Este procesado asume un comportamiento "lento" en cuánto a las intervenciones de locutores durante un flujo de audio se refiere. Es decir, trata de eliminar cambios rápidos en la

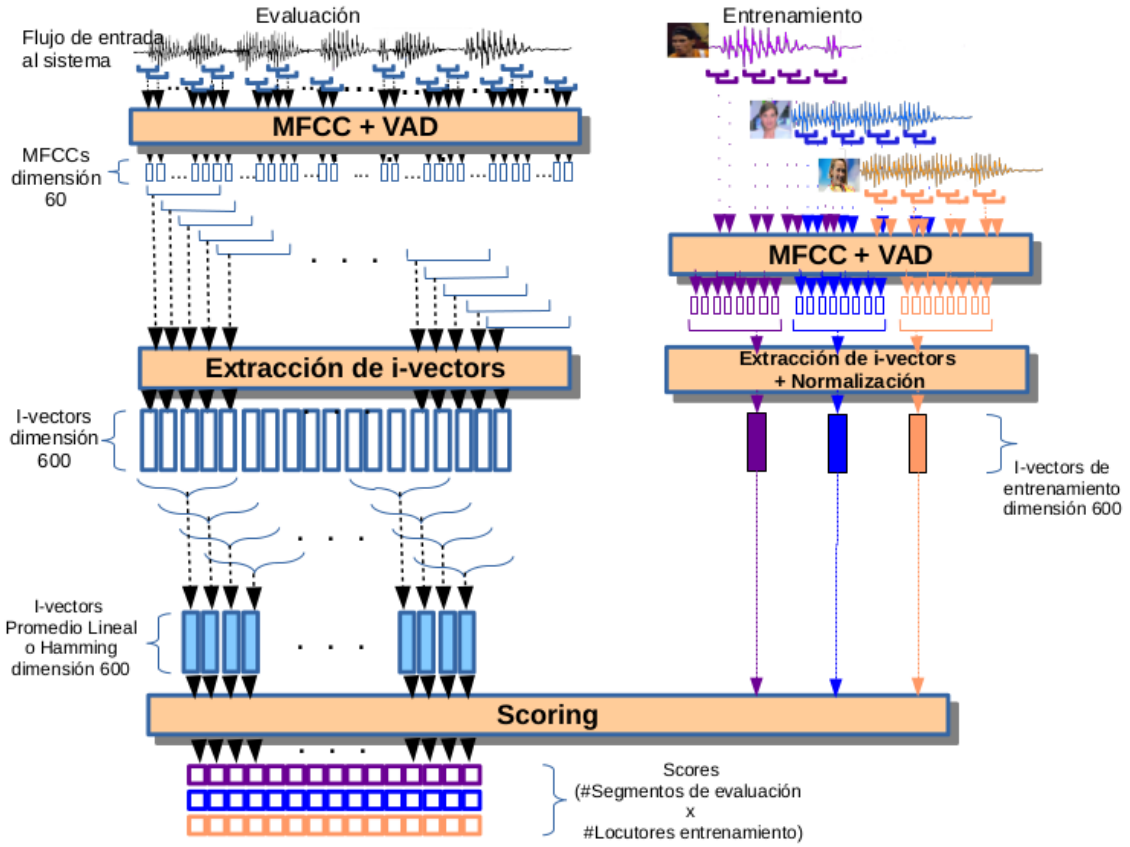

Figura 4.3: Esquema explicativo del proceso de promediado de i-vectors (Para N-ivectors promedio).

decisión del sistema, que prevesiblemente tienden a ser poco naturales y posiblemente tengan que ver con una mala aproximación de los i-vectors.

Además, este técnica permite incorporar información proveniente de i-vectors de entrenamiento y evaluación (puntuación proveniente de ambos). En lugar del promediado de i-vectors, que únicamente aporta información que provenga del audio de evaluación.

Se aplican dos aproximaciones posibles: promediado lineal y promediado Hamming. Donde se aplican diferentes ponderaciones en función del tipo de promediado de la siguiente forma:

$$s_{promedio}(n) = s_(n) \cdot v(n) \tag{4.5}$$

donde v se define como:

$$v_{rectangular}(n) = \begin{cases} 1, & -\frac{N-1}{2} < n < \frac{N-1}{2} \\ 0, & resto \end{cases}$$

para promediado lineal,

$$v_{hamming}(n) = 0{,}54 - 0{,}46\cos\left(2\pi\frac{n}{N}\right), \quad -\frac{N-1}{2} \leq n \leq \frac{N-1}{2} \tag{4.6}$$

para promediado Hamming.

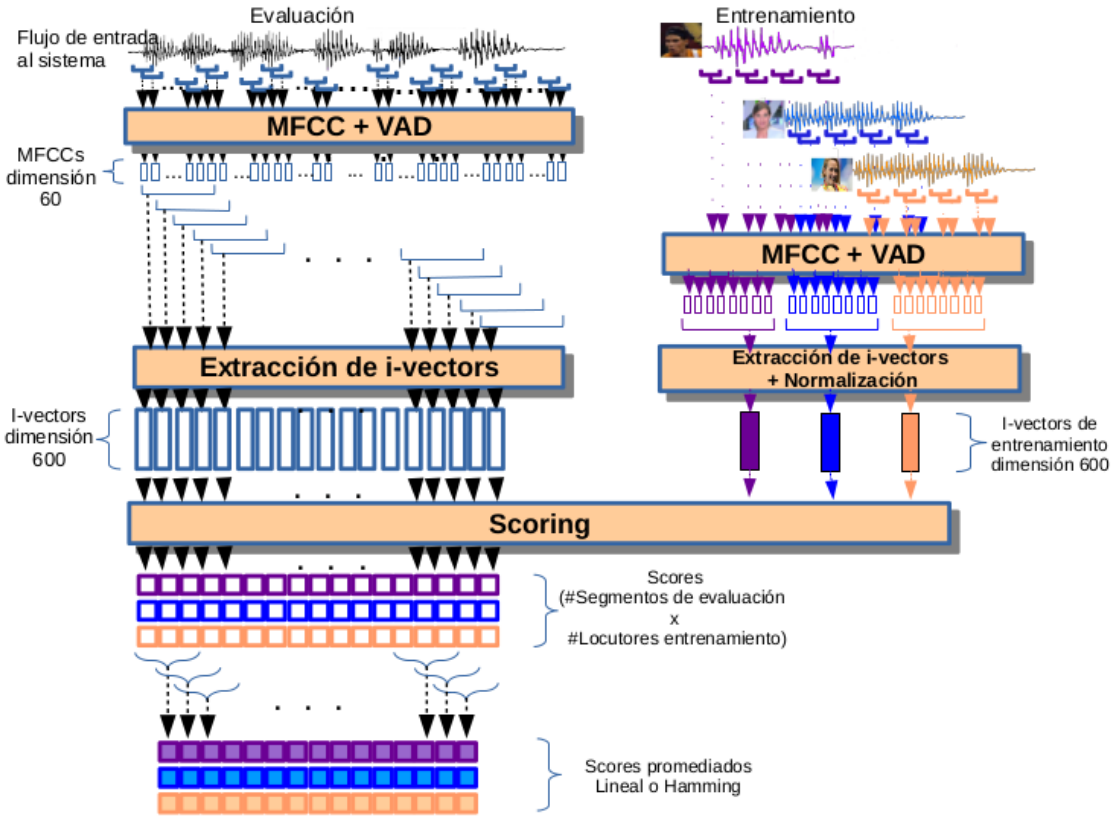

Figura 4.4: Esquema explicativo del proceso de promediado de *scores* (Para N-*scores* promedio).

4.2.4. Concatenación de locuciones de entrenamiento

Una de las prácticas más extendidas en los sistemas de reconocimiento de locutor, se basa en cálculo de un i-vector medio utilizando todos los i-vectors de entrenamiento para cada uno de los locutores a identificar.

Teniendo en cuenta la baja duración de las locuciones en el dominio de audio *broadcast* y suponiendo que los i-vectors de corta duración son una versión ruidosa de un i-vector de mayor longitud.

Se propone utilizar la concatenación de locuciones cortas y posterior extracción de un único i-vector de entrenamiento por locutor. En lugar del promediado de i-vectors provenientes de locuciones "largas" en un único i-vector de entrenamiento, como se viene haciendo habitualmente en el estado del arte actual.

4.3. Sistema Adaptación de Dominio

Las técnicas presentes en los sistemas de reconocimiento de locutor clásicos (como en este caso) son altamente dependientes de la cantidad de datos etiquetados y no etiquetados disponibles. Para realizar el entrenamiento de hiper-parámetros (UBM, T, m, W, AC, WC, ver Figura 3.1) es necesario disponer de centenares de locutores con decenas de locuciones por cada uno de ellos. Este escenario es altamente irrealista en cuanto a la cantidad de

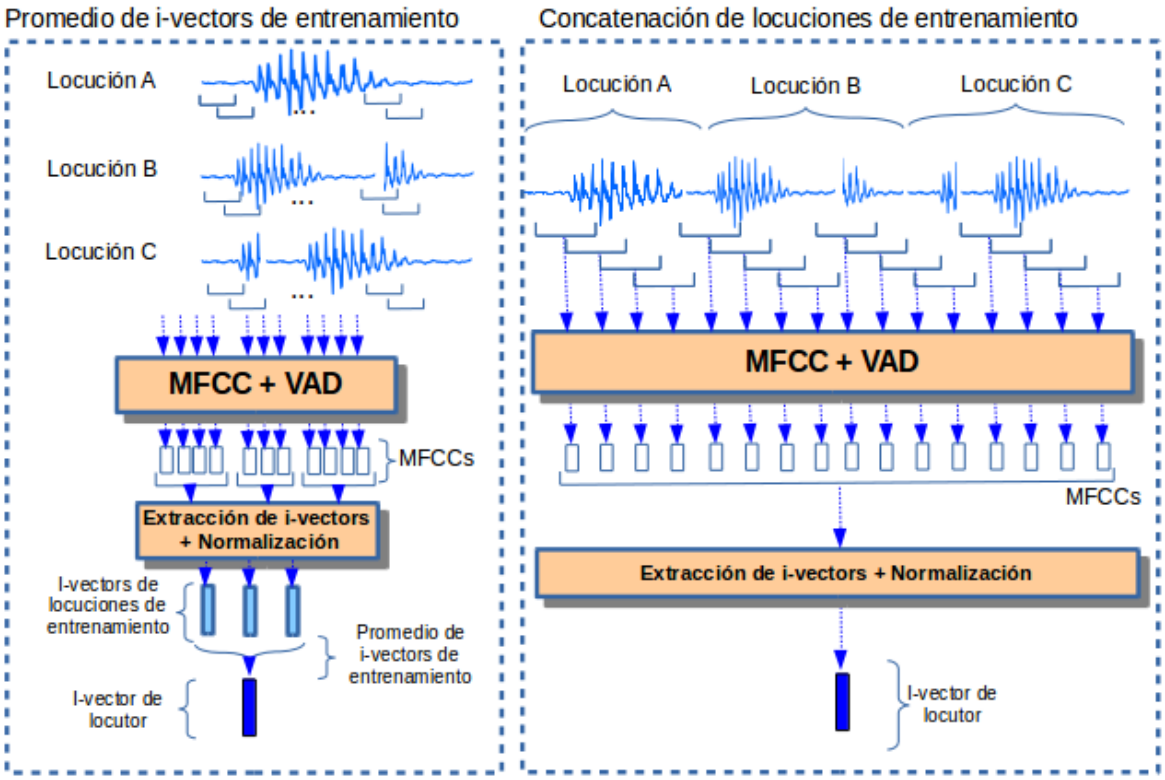

Figura 4.5: Esquema comparativo entre de cálculo de i-vector medio por locutor (entrenamiento) a la izquierda y concatenación de locuciones con posterior extracción de un único i-vector de locutor (entrenamiento) a la derecha.

datos disponibles para el entrenamiento de un sistema de reconocimiento de locutor en audio *broadcast*.

El sistema *baseline* hasta ahora desarrollado ofrece un entrenamiento de hiper-parámetros a partir de datos provenientes de un entorno muy específico y muy distinto al entorno de la aplicación concreta para la que se quiere utilizar. Los datos de entrenamiento provienen de un dominio principalmente en idioma inglés y voz telefónica. En cambio, se trata de utilizar para el reconocimiento de voz microfónica e idioma castellano.

Debido a este problema (pocos datos en el nuevo dominio), se aplican técnicas de Adaptación de Dominio, con el fin de adaptar el sistema al nuevo entorno con utilizando una pequeña cantidad de datos (Audias-Radio 2015).

Según las técnicas presentadas en la sección 3.2.1, las principales mejoras en el uso de técnicas de adaptación de domino provienen de la adaptación del bloque PLDA (AC y WC). El resto de hiper-parámetros tales como UBM y T, en principio, no aportan una mejora sustancial en el rendimiento del sistema, pero de igual manera se valorará cuantitativamente su impacto en la sección de experimentos y resultados, ver sección 6.3.1.

4.3.1. Entrenamiento de hiper-parámetros con distintos conjuntos de datos.

Con el objetivo de validar el grado de importancia y la variación en el rendimiento según la cantidad de datos disponibles, se entrena el sistema con distintas configuraciones en cada

una de sus etapas.

Variando los datos de entrenamiento en cada uno de los hiper-parámetros que afectan al sistema, se pretende observar y comprobar cuales son las etapas críticas (UBM, T, normalización, PLDA, etc.) en función del rendimiento del sistema, a partir de los conjuntos de datos disponibles, ver Figura 4.6.

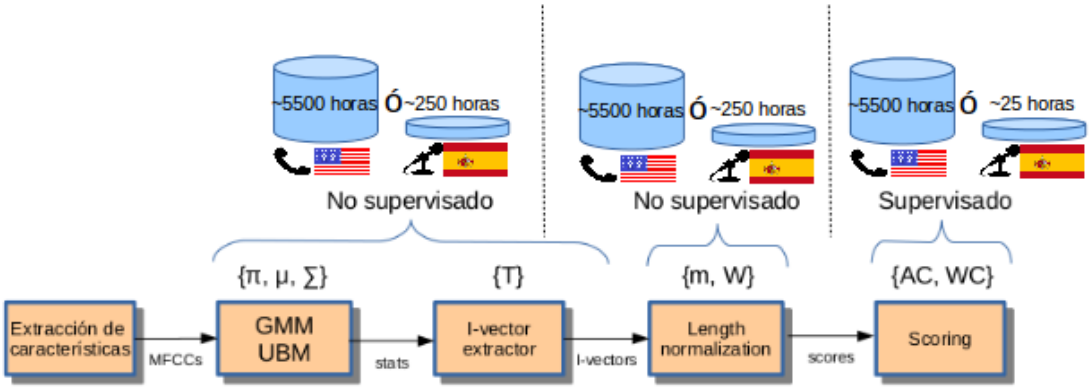

Figura 4.6: Esquema con distintas posibilidades de entrenamiento para hiper-parámetros, según los subconjuntos de datos disponibles.

Esto permitirá observar la diferencia entre un sistema completamente entrenado con datos de desarrollo que provengan de gran cantidad de datos *out-domain*, frente al mismo sistema entrenado con poca cantidad de datos *in-domain*. Mostrando una diferencia sustancial de rendimiento debido a la cantidad de datos disponibles, ver sección 6.3.1. Derivado de ello, se propondrán diferentes alternativas con el objetivo de adaptar ciertas etapas haciendo uso de datos de ambos dominios.

4.3.2. Adaptación de Dominio Supervisada

En particular, las mejoras sustanciales en el rendimiento de los sistemas provienen de una adaptación sobre PLDA. Dicho bloque de entrenamiento implica el uso de datos previamente etiquetados (*supervised learning*), por tanto, será indispensable utilizar la parte etiquetada de Audias-Radio como base de datos *in-domain* para realizar dicha adaptación.

Interpolación de matrices de covarianza PLDA

Para dicha adaptación se propone un enfoque basado en la interpolación de los parámetros de PLDA (matrices de covarianza inter e intra-clase). Para ello, se aplica interpolación de matrices de covarianza inter-clase (Σ ó AC) e intra-clase (Φ ó WC) entre ambos dominios *out-domain* (telefonico-ingles) e *in-domain* (microfónico-castellano) de la siguiente manera:

$$\Phi_{adapt} = \alpha\Phi_{in-domain} + (1-\alpha)\Phi_{out-domain} \tag{4.7}$$

$$\Sigma_{adapt} = \alpha\Sigma_{in-domain} + (1-\alpha)\Sigma_{out-domain} \tag{4.8}$$

donde α se presenta como el coeficiente de adaptación/interpolación que permite balancear al peso de cada conjunto de datos, variando su valor entre 0 y 1.

Esta interpolación permite realizar una adaptación en la estimación de variabilidad inter e intra-clase entre los datos *out-domain* e *in-domain*. Por lo que, aporta información presente en el nuevo dominio (*in-domain*), sobre un entrenamiento previo de matrices de covarianza con gran cantidad de datos (*out-domain*).

Con ello se pretende mejorar el rendimiento del sistema, al hacer una mejor estimación de la variabilidad de los datos. Que debería traducirse en una etapa PLDA mejor adaptada a los datos de evaluación y por tanto, esto implicaría una mejora sustancial en el rendimiento, [Garcia-Romero and McCree, 2014].

4.3.3. Adaptación de Dominio No Supervisada

Con el objetivo de aprovechar gran parte de los datos provenientes de Audias-Radio 2015 (*in-domain*) no etiquetados, que suponen la gran mayoría de datos (250 horas). Se parte de una nueva alternativa para realizar una adaptación de PLDA a partir de datos no etiquetados.

Según estudios del estado del arte, este tipo de técnicas permiten mejorar en gran medida resultados obtenidos con aprendizaje supervisado, debido al hecho de utilizar mayor número de datos, [Garcia-Romero et al., 2014].

Para ello se aplica un VAD (*Voice Activity Detector*) que permite extraer las locuciones del conjunto de 250 horas no etiquetadas y a partir de estas locuciones se realiza un *clustering* de datos que permita realizar un "etiquetado automático" de segmentos de voz con distintos locutores.

AHC + Interpolación de matrices de covarianza PLDA

El método AHC (*Agglomerative Hierarchical Clustering*)es un método de *clustering* jerárquico que permite agrupar muestras, en este caso i-vectors, con cierta similitud entre ellas, en base a una métrica definida (en este caso distancia coseno), ver Figura 4.7.

El método consiste en comenzar con cada i-vector como un *cluster* separado y en cada paso o iteración del algoritmo, se fusionan los dos *clusters* más cercanos, basado en una métrica predefinida (distancia coseno). Este algoritmo de fusión permite definir una ruta en el espacio de las particiones. Obteniendo una agrupación final basada en un criterio de detención (*cutoff*), [Duda et al., 2001]. Este criterio de detención se ajustará de forma empírica a partir de los datos etiquetados. Para ello, se aplicará AHC sobre la parte de Audias-Radio-2015 etiquetada y se procederá a realizar un barrido de valores sobre el criterio de detención que permita obtener su valor óptimo. En la sección 6.3.3 se realizará una explicación más detallada sobre este procedimiento.

Este etiquetado estimado, permite aplicar de nuevo la técnica de interpolación de matrices de covarianza en PLDA. En esta ocasión la cantidad de datos para entrenar las matrices con datos *in-domain* aumenta considerablemente (de 25 a 250 horas). Esto hace posible una mejora sustancial en su estimación y por tanto, permitiría mejorar el rendimiento del sistema propuesto con adaptación de dominio supervisado (*supervised domain adaptation*).

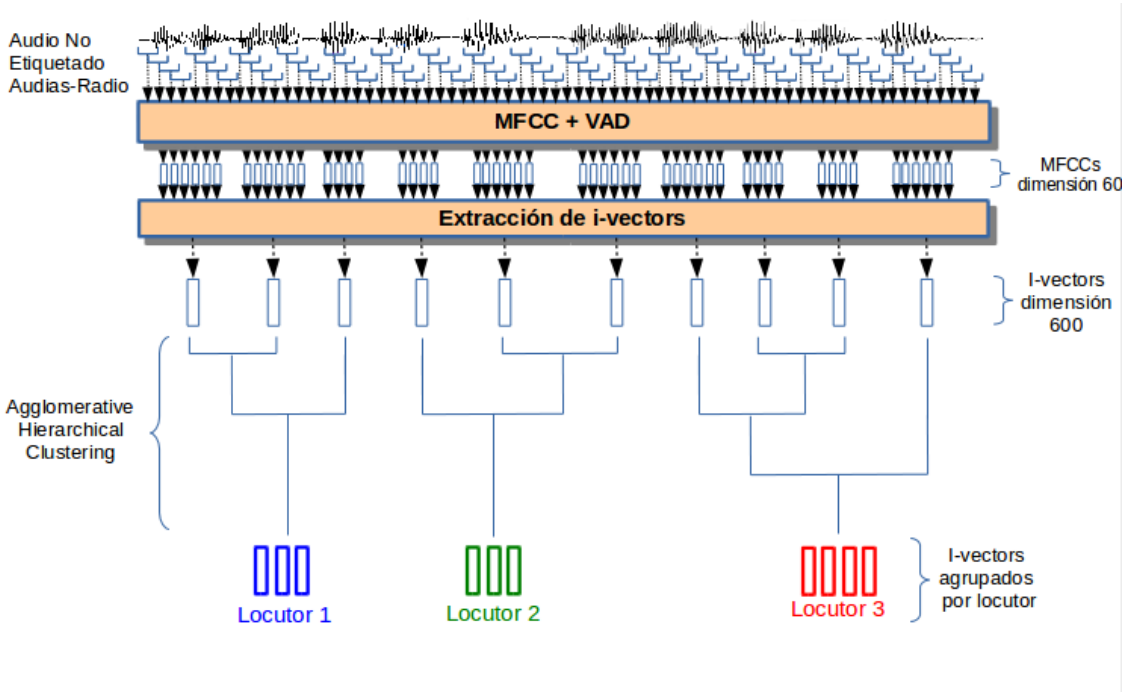

Figura 4.7: Esquema ejemplo de etiquetado automático mediante el algoritmo AHC, con 10 i-vectors y 3 locutores etiquetados.

Capítulo 5

Entorno experimental

Este capítulo describe el software y las diferentes bases de datos utilizadas para desarrollo, entrenamiento y evaluación. Así como el protocolo con el que se evalúan los sistemas propuestos anteriormente en el Capítulo 4.

5.1. Software Kaldi

Durante el desarrollo y la implementación de todo el proyecto se ha trabajado con el software Kaldi [Povey et al., 2011].

Este software se trata de un *toolkit* desarrollado en C++ específicamente diseñado para su uso por la comunidad científica en tareas de procesado de voz (reconocimiento de locutor inclusive).

Principalmente provee códigos flexibles que ofrecen gran versatilidad en la implementación de métodos ya existentes o prototipado rápido. Para ello implementa funciones a bajo nivel (C++) y scripts con implementaciones básicas de algoritmos de procesado de voz en lenguajes de programación Perl, Bash y Python principalmente. Esto permite realizar diseños extensibles, utilizando algoritmos genéricos en funciones de alto nivel, para la implementación de sistemas específicos.

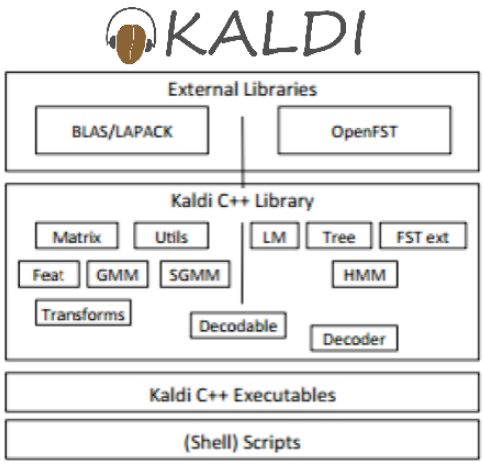

Figura 5.1: Estructura general del software Kaldi.

Este software se encuentra bajo licencia Apache v2.0 y tiene como dependencia varias librerías externas: OpenFST para trabajar con integración a nivel de código y "*Basic Linear Algebra Subrutines*" (BLAS) o "*Linear Algebra PACKage*" (LAPACK) como soporte de álgebra numérico.

5.2. Bases de datos

Para la realización de este proyecto se dispondrán de tres bases de datos: NIST Speaker Recognition Evaluation Database (SRE), Switchboard (SWBD) y Audias-Radio-2015, ver Figura 5.2. Las dos primeras de ellas se utilizarán como una única base de datos conjunta, debido a su similitud en cuanto al tipo de audio que contienen (voz microfónica en idioma inglés) y formarán el conjunto de datos *out-domain* (fuera de dominio). Por otro lado, Audias-Radio-2015 se compone de dos partes, donde solamente una de ellas dispone de audio etiquetado, ver Figura 5.2.

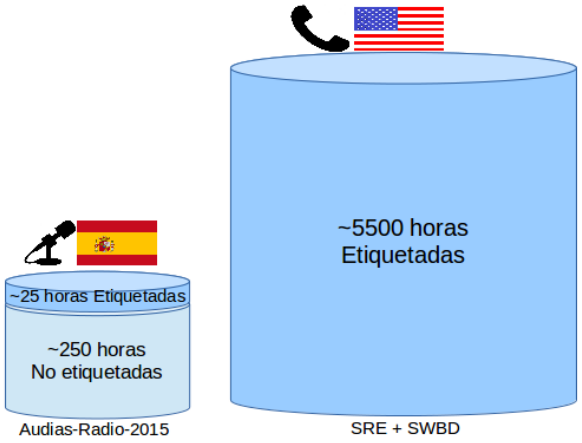

Figura 5.2: Representación de bases de datos disponibles para cada dominio y partición de datos etiquetados y no etiquetados.

A continuación se detallarán las características específicas de cada una de ellas.

5.2.1. NIST Speaker Recognition Evaluation Databases - SRE

Speaker Recognition Evaluations (SRE) se compone de una serie de evaluaciones llevadas a cabo anualmente por NIST (*US National Institute of Standards and Technology*) que componen una importante contribución en cuanto a investigación se refiere. Con intención de ser de interés para todos los investigadores que trabajan en reconocimiento de voz independiente del texto. La evaluación está diseñada para ser simple, enfocarse en los problemas principales de la tecnología y contar con un soporte completo y accesible que permita medir el estado del estado del arte, en un marco común para todos los investigadores.

En este marco de trabajo se encuentra una parte de las bases de datos utilizadas durante este proyecto. Dichas bases de datos consisten en dos subconjuntos: uno de entrenamiento (*train*) donde se conoce a que locutor pertenece cada locución y un segundo conjunto de evaluación (*test*) que contiene locutores totalmente desconocidos. En ambos subconjuntos se incluyen diferentes condiciones de duración, canal, género, etc. Que permitirán entrenar y evaluar el sistema en un entorno lo más genérico posible.

En particular, para la realización de este proyecto se han seleccionado corpus de SRE04, 05, 06 ,y 08.

2004 NIST Speaker Recognition Evaluation

Este conjunto de datos contiene locuciones habla conversacional telefónica proveniente de diálogos multi-canal recopilados simultáneamente de una serie de micrófonos auxiliares. Presenta alrededor de 700 locutores distintos, con un total de 4600 locuciones aproximadamente que suman un total de una 280 horas de audio.

2005 NIST Speaker Recognition Evaluation Training Data

Los datos de voz consisten en habla conversacional telefónica proveniente de diálogos multicanal recopilados simultáneamente de una serie de micrófonos auxiliares. Los archivos están organizados en dos tipos: extractos de dos segundos de 10 segundos (segmentos continuos de conversaciones individuales que se estima que contienen aproximadamente 10 segundos de voz real en el canal de interés) y conversaciones de 2 a 5 minutos. Presenta alrededor de 700 locutores distintos, con un total de 2800 locuciones aproximadamente que suman un total de una 230 horas de audio.

2006 NIST Speaker Recognition Evaluation Training Set

Los datos de voz en esta versión fueron recopilados por LDC como parte del proyecto Mixer, en particular las fases 1, 2 y 3 de Mixer. El proyecto Mixer admite el desarrollo de tecnología robusta de reconocimiento de locutor al proporcionar un discurso cuidadosamente recopilado y auditado de una gran cantidad de locutores grabados simultáneamente a través de numerosos micrófonos y en diferentes situaciones comunicativas en múltiples idiomas. Por tanto, incluye gran cantidad de variabilidad de canal y de dispositivos de grabación haciendo que sea una base de datos mucho más realista. Los datos en su mayoría son en inglés, pero incluyen algunas conversaciones en árabe, bengalí, chino, hindi, coreano, ruso, tailandés y urdu.

Los segmentos de voz telefónica corresponden con conversaciones multi-canal recogidas simultáneamente de una serie de micrófonos auxiliares. Los archivos están organizados en tres tipos: extractos de dos canales de aproximadamente 10 segundos, conversaciones de dos canales de aproximadamente 5 minutos y conversaciones de canales sumados de aproximadamente 5 minutos.

Presenta alrededor de 2200 locutores distintos con un total de 18000 locuciones aproximadamente, sumando un total de unas 1500 horas de audio.

2008 NIST Speaker Recognition Evaluation Training Set

Los datos de voz en fueron recopilados en 2007 por LDC formando parte de proyecto *Mixer 5*, con condiciones similares a los datos recogidos Mixer 1, 2 y 3. Principalmente, se compone de 523 de habla telefónica conversacional predominantemente en inglés (aunque incluye otros idiomas) y 427 horas de habla microfónica en formato entrevista, totalmente en inglés americano.

Los segmentos de voz telefónica incluyen segmentos en el rango de 8-12 segundos y 5 minutos de conversaciones más largas. Los datos de entrevistas incluyen segmentos cortos de

conversación de aproximadamente 3 minutos de una sesión de entrevista más larga. Al igual que en las evaluaciones anteriores, los intervalos de silencio no se eliminaron. Además, se proporcionan dos canales de conversación separados (para ayudar a los sistemas en cancelación de eco, análisis de diálogo, etc.).

5.2.2. Switchboard - SWBD

Esta base de datos corresponde a un corpus recopilado por el *Linguistic Data Consortium* (LDC) en apoyo a un proyecto de reconocimiento de locutor del Departamento de Defensa de los EE. UU. Principalmente se caracteriza por contener llamadas telefónicas con temáticas previamente predefinidas y en idioma inglés.

Switchboard-2 Phase-II

La denominada *SWB-2 Phase II* consiste en 4472 conversaciones telefónicas de 5 minutos de duración, con un total de 679 locutores que hacen un total de 445 horas de audio. Siendo todas ellas en idioma inglés, con interlocutores residentes en campus universitarios del medio-oeste de EE. UU..

Todas las grabaciones fueron auditadas por miembros del de LDC. Prestando especial atención a la verificación del locutor, la duración de la llamada y la calidad.

Switchboard-2 Phase-III

SWB-2 Phase-3 corresponde con un total de 2.728 llamadas, o 5.456 canales, de duración comprendida entre 5 y 6 minutos, y 640 participantes (292 hombres, 348 mujeres), en diversas condiciones (calidad de canal, teléfonos, ruido ambiente, etc.). Enfocada principalmente en el sur de EE. UU. con conversaciones totalmente en idioma inglés. Incluyendo un alrededor de 5500 locuciones que forman un corpus con 400 horas de audio neto.

Switchboard Cellular-2

SWB Cellular Part 2, desarrollada por el Linguistic Data Consortium (LDC), consta de aproximadamente 200 horas de conversaciones telefónicas en inglés sobre un canal de transmisión CDMA. Consta de un total de 2,020 llamadas de telefonía movil, o 4,040 canales, con una duración entre 5 y 6 minutos y realizadas por 419 locutores (2,405 mujeres, 1,635 hombres) en diversas condiciones (calidad de canal, teléfonos, ruido ambiente, clipping, etc.).

5.2.3. Audias-Radio 2015

Dada la necesidad de desarrollar e implimentar un sistema de detección de locutores en audio *broadcast*, es necesario el uso de un corpus de datos en ese dominio específico. Para ello, se cuenta con Audias-Radio 2015, extraída en su totalidad de programas radiofónicos españoles con un total de 250 horas de audio calidad *broadcast*. Distribuyéndose en 25 horas etiquetadas a nivel de locutor (válidas para aprendizaje supervisado) y 250 sin etiquetar locutores (válidas para aprendizaje no supervisado). Este corpus de datos se caracteriza por alta variabilidad en las condiciones de grabación tanto a nivel de instrumentación (conexiones telefónicas, micrófonos, etc.), acústicas (mitines, estudio de grabación, ruedas de prensa, etc), solapamiento (locutor-locutor, música-locutor), duraciones (breves declaraciones, noticiario, monólogos extensos, etc.), sonidos particulares (señales horarias, risas "enlatadas", etc.).

A continuación se incluye una descripción detallada de las estadísticas de más relevantes en la base de datos:

Dataset	Audias-Radio-2015 Etiquetada	Audias-Radio-2015 Sin etiquetar
#Locuciones	4541	-
#Ficheros	50	250
#Locutores	245	-
Duración total	25h	250h
Max/Min/Media #Locuciones/Locutor	611 / 1 / 18	-
Max/Min/Media #Duración/Locución	5m 11s / 1s / 12s	-
Max/Min/Media #Duración/Locutor	1h 21m 35s / 1s / 3m 45s	-
Idioma	Castellano	Castellano
Fuente	Broadcast	Broadcast

Tabla 5.1: Descripción detallada base de datos Audias-Radio-2015.

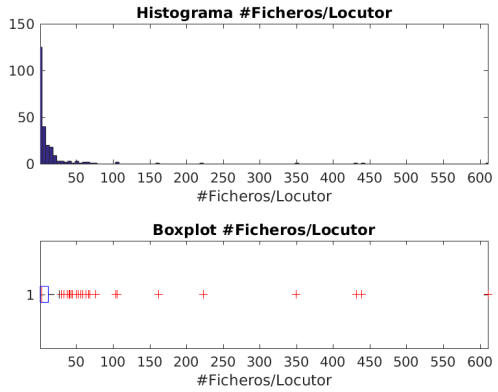

Figura 5.3: Histograma y boxplot del número de locuciones por locutor en Audias-Radio-2015.

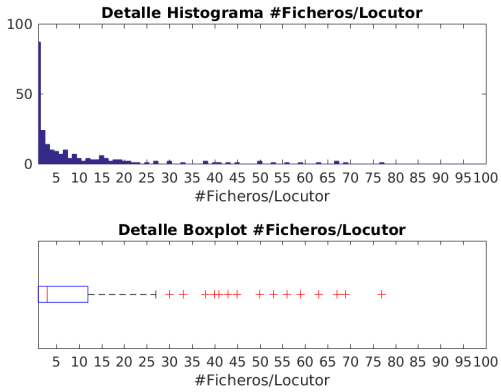

Figura 5.4: Histograma y boxplot en detalle del número de locuciones por locutor en Audias-Radio-2015.

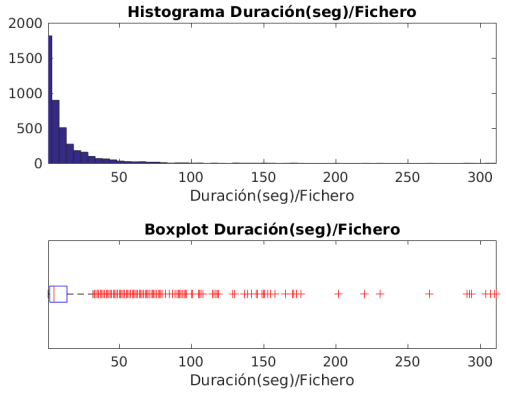

Figura 5.5: Histograma y boxplot de duración por locución en Audias-Radio-2015.

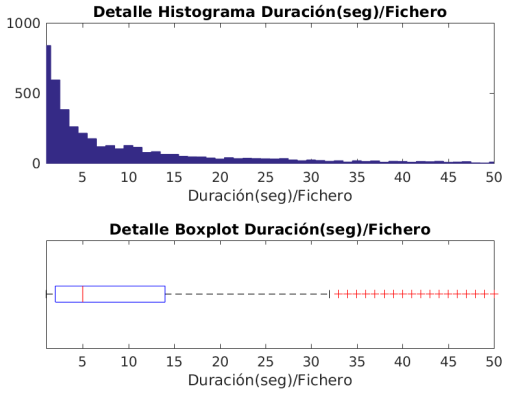

Figura 5.6: Histograma y boxplot en detalle de duración por locución en Audias-Radio-2015.

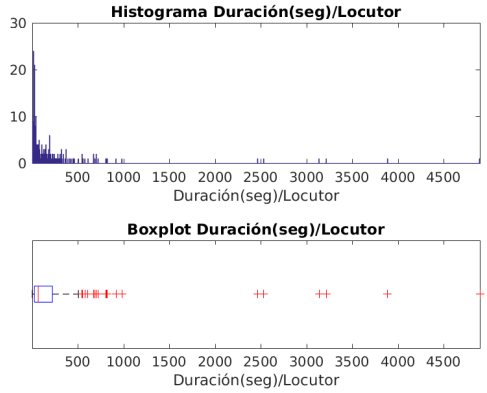

Figura 5.7: Histograma y boxplot de duración por locutor en Audias-Radio-2015.

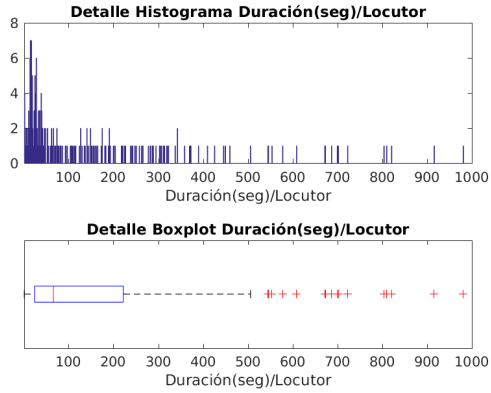

Figura 5.8: Histograma y boxplot en detalle de duración por locutor en Audias-Radio-2015.

La recopilación de datos y su posterior etiquetado se realizó durante 2015 por parte de miembros del laboratorio de investigación AUDIAS - Audio, Data Intelligence and Speech, [Fernández Gallego, 2016][Soriano Morancho et al., 2016][García Naranjo et al., 2016] [Escudero Barrero et al., 2016]. Por tanto, se considera una marco adecuado (base de datos puramente *broadcast*) en el proceso de entrenamiento y evaluación del sistema propuesto.

5.3. Particionado del conjunto de datos

El buen funcionamiento de los sistemas de reconocimiento automático pasa por seleccionar adecuadamente los datos a utilizar en los distintos bloques que componen el sistema. Por ello, se debe hacer uso de todos los datos disponibles en las bases de datos descritas anteriormente de la forma más óptima posible.

Una mayor cantidad de datos de entrenamiento permitirá construir sistemas más robustos que capten la variabilidad de los datos y modelen mucho mejor las diferentes condiciones que se puedan presentar. Por otro lado, una gran cantidad de datos de evaluación permitirá construir un entorno mucho más realista que simule de mejor manera las condiciones presentes en un determinado dominio.

Es en este punto donde la limitación que viene dada por la cantidad de datos disponibles debe ser tratada de forma eficiente. Teniendo en cuenta la cantidad de datos necesarios para el correcto entrenamiento de cada una de las partes involucradas en el sistema. Por tanto, se extraerán y dividirán los datos de la siguiente manera:

- Datos de desarrollo (*development dataset*), utilizado principalmente para entrenamiento de UBM, T, normalización y PLDA:

Datos de desarrollo				
Dataset	#Locutores	#Locuciones	Duración total (horas)	% de aportación al conjunto
2004 NIST SRE	307	4577	380	9 %
2005 NIST SRE Training Data	719	2764	230	5 %
2006 NIST SRE Training Set	2227	18320	1526	36 %
2008 NIST SRE Training Set	1320	10973	914	22 %
TOTAL SRE	3805	36612	3050	72 %
Switchboard-2 Phase-II	679	8939	445	11 %
Switchboard-2 Phase-III	640	5314	412	10 %
Switchboard Cellular-2	418	4038	304	7 %
TOTAL SWB	1737	18291	1161	28 %
TOTAL	5542	54903	4211	100 %

Tabla 5.2: Descripción detallada de datos para desarrollo, NIST SRE y Switchboard.

- Datos de entrenamiento (*train dataset*), utilizados para crear los modelos de los locutores a identificar: (*Estos datos son orientativos y se describirán específicamente para cada una de las pruebas realizadas en el Capítulo 6).

Datos de entrenamiento			
Dataset	#Locutores	#Locuciones por locutor	Duración por locución
Subset Audias-Radio-2015	22-64	3-6	5-10 segundos

Tabla 5.3: Descripción de la partición de datos para entrenamiento.

- Datos de evaluación (*test dataset*), compuesto por flujos continuos de audio (programas radiofónicos completos de 30 minutos de duración) cuyo principal objetivo es evaluar el rendimiento del sistema:

Datos de evaluación			
Dataset	#Locutores	#Programas	Duración total
Subset Audias-Radio-2015	22-64 target (223-181 non-target)	50	25 horas

Tabla 5.4: Descripción de la partición de datos para evaluación.

5.4. Evaluación del rendimiento

Habitualmente los sistemas de reconocimiento de locutor tienen como salida una medida de similitud entre dos muestras dadas (*trial*), denominada puntuación o *score*. Con el objetivo de realizar una clasificación es necesario seleccionar un umbral (*threshold*) que haciendo uso de estos *scores* permita clasificar entre muestras que pertenecen al mismo individuo y a individuos diferentes.

Por tanto, los *trials* se clasifican como *target*, cuando las muestras corresponden al mismo individuo, y *non-target*, cuando corresponden a individuos diferentes. Estos *trials* deberán ser aceptados o rechazados utilizando un umbral que permite decidir que *trials* son clasificados como *target* (aceptados) y como *non-target*(rechazados).

Es por esto que se definen dos tipos de posibles sucesos a la salida:

- FA (*false aceptance* o *false positive*): *trial non-target* es aceptado por el sistema.

- FR (*false rejection* o *false negative*): *trial target* es rechazado por el sistema.

Para un umbral fijado dado un conjunto de *trials* y sus respectivos *scores*, es posible determinar la probabilidad de FA (pFA) y FR (pFR) de la siguiente manera:

$$P_{FA} = \frac{total \quad FA \quad trials}{total \quad non-target \quad trials} \cdot 100 \tag{5.1}$$

$$P_{FR} = \frac{totalFRtrials}{total \quad target \quad trials} \cdot 100 \tag{5.2}$$

Realizando un barrido del umbral es posible determinar el comportamiento del sistema en todo el rango de *scores*. Por tanto, las curvas DET (*Detection Error Tradeoff*) permiten visualizar pFA frente pFR para todo punto de operación del sistema:

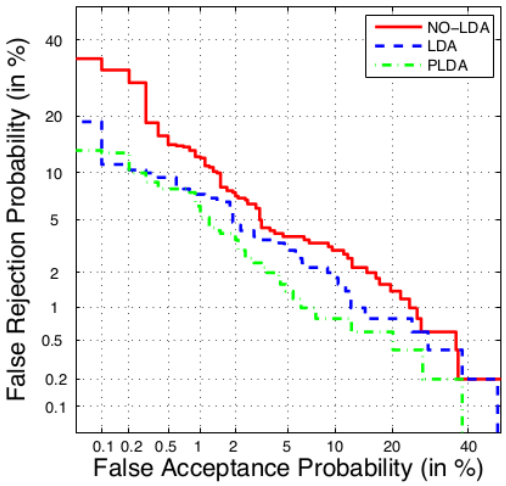

Figura 5.9: Ejemplo de curva DET.

Siendo el EER (*Equal Error Rate*) el punto de operación del sistema donde pFA = pFR.

Es por tanto, que durante este proyecto se utilizarán como medidas de rendimiento tanto curvas DET como sus EERs asociados.

Eliminación de fronteras en zonas voz/no-voz durante la evaluación del sistema

Las muestras de evaluación son extraídas de un flujo continuo de audio procedente de programas radiofónicos. Con el objetivo de identificar cada locutor en un segmento corto de tiempo, se procede a extraer i-vectors en segmentos de 5 segundos con solapamiento 1 segundo entre ellos. Es por ello, que existen ciertas zonas al inicio y fin de locución, denominadas frontera, donde existe una clara ambigüedad acerca de si corresponden con un locutor, ruido, solapamiento entre locutores, etc. Por tanto, estas zonas (fronteras) serán descartadas en el proceso de evaluación del sistema, con el fin de obtener unos resultados mucho más ajustados al rendimiento real en segmentos claramente definidos como voz o no-voz

Teniendo en cuenta esto, se plantea un esquema de eliminación de fronteras del siguiente modo:

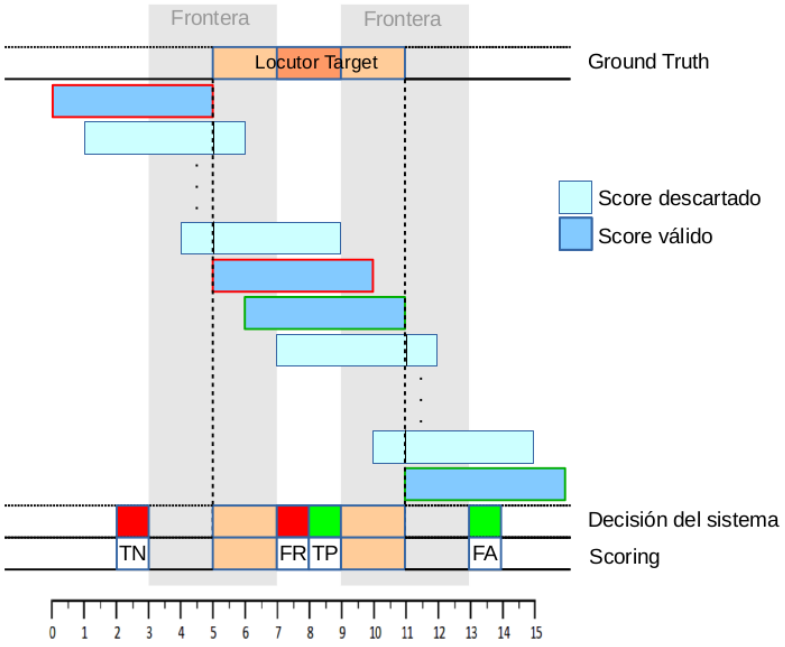

Figura 5.10: Esquema de supresión de fronteras entre segmentos de voz y no voz. Donde los *scores* que incluyen información de la zona frontera (zona gris) son eliminados de la evaluación. Descripción de las distintas situaciones en la decisión del sistema; verdadero negativo (TN), falso rechazo (FR), verdadero positivo (TP) y falsa aceptación (FA).

Por tanto, todos aquellos *scores* que contengan información de audio presente en la zona de frontera serán descartados de la evaluación, incluyendo los *scores* promediados o i-vectors promediados en los experimentos correspondientes. Obteniendo diferentes longitudes de frontera por experimento:

- 5 *scores* descartados por frontera, en sistema de locuciones cortas sin promediado (raw).
- 7 *scores* descartados por frontera, en sistema de locuciones cortas con promediado de 3 scores/i-vectors.
- 9 *scores* descartados por frontera, en sistema de locuciones cortas con promediado de 5 scores/i-vectors.
- 11 *scores* descartados por frontera, en sistema de locuciones cortas con promediado de 7 scores/i-vectors.

Esto hace, que los resultados obtenidos en cuanto a rendimiento del sistema estén sesgados por la duración de las locuciones. Debido a la eliminación de fronteras, un aumento del promediado de *scores*/i-vectors, puede eliminar de la evaluación segmentos entre 5 y 10 segundos, dependiendo de la longitud del promediado.

Capítulo 6

Experimentos y resultados

En este capítulo se presentan los experimentos desarrollados durante la evaluación y desarrollo de los sistemas descritos en el Capítulo 4. Esto nos permitirá evaluar y ajustar las técnicas propuestas en la detección de locutores a partir de locuciones cortas (alrededor de 5 segundos) y flujos continuos de audio (50 programas radiofónicos). Así como, la evaluación del sistema propuesto para la tarea de Adaptación de Dominio con los conjuntos de datos disponibles en distintos dominios (calidad microfónica-inglés y calidad radiofónica-castellano).

Los experimentos detallados a continuación tienen como objetivo determinar el impacto sobre el rendimiento de cada una de las técnicas y aproximaciones propuestas en los sistemas del Capítulo 4. Esto nos permitirá evaluar el rendimiento del sistema sobre cada uno de los objetivos propuestos (mejoras en locuciones cortas y mejoras en adaptación de dominio) con el fin de desarrollar un sistema completo que permita obtener el mejor rendimiento posible con los datos disponibles para la tarea.

Para ello, se propone la evaluación del sistema en dos versiones a partir de la cantidad de datos presente en Audias-Radio-2015:

- Versión optimista: Se tomarán **22 locutores** a detectar por el sistema. Correspondientes con aquellos locutores que presentan un número de locuciones igual o superior a 10 y una duración igual o superior a 10 segundos por locución. **Utilizando 6 locuciones de al menos 10 segundos para el entrenamiento de sus respectivos modelos**.
- Versión pesimista: Se tomarán **64 locutores** a detectar por el sistema. Correspondientes con aquellos locutores que presentan un número de locuciones igual o superior a 5 y una duración igual o superior a 5 segundos por locución. **Utilizando 3 locuciones de 5 segundos para el entrenamiento de sus respectivos modelos**.

6.1. Sistema *baseline*

En primer lugar, y con el objetivo de tomar un punto de referencia en el análisis del rendimiento de los sistemas, se ha evaluado el sistema *baseline* descrito en el Capítulo 4.

Esta evaluación se propone en el marco de un sistema clásico de reconocimiento de locutor. Para ello, habitualmente, los sistemas se evalúan a partir de locuciones previamente segmentadas, donde está claramente definido su inicio y fin. Además, normalmente se realizan únicamente comparativas entre segmentos de audio con voz, es decir, no se realizan intentos de identificación entre un locutor y segmentos de música, ruidos, silencios, etc. Por tanto,

durante la evaluación de este sistema se realizará la comparativa entre locuciones provenientes del mismo o diferente locutor, pero siempre se corresponderán con locuciones reales y segmentadas manualmente.

Cabe destacar, el uso exclusivo de las locuciones utilizadas en entrenamiento con el único fin de generar los modelos de locutor. Eliminando por tanto, dichas locuciones del conjunto de evaluación para evaluar el rendimiento del sistema.

La tabla 6.1 muestra los resultados del sistema en sus dos versiones (22 y 64 locutores), aplicando normalización de *scores* (z-norm) y distintas técnicas en el proceso de puntuación (*cosine scoring*, LDA y PLDA):

Sistema		EER(%)		
Locutores	Normalización	Cosine	LDA	PLDA
22	-	6.25	3.71	3.30
22	z-norm	5.39	3.01	**3.13**
64	-	7.23	5.59	3.89
64	z-norm	5.82	4.10	**3.59**

Tabla 6.1: Rendimiento del sistema *baseline* en EER(%)

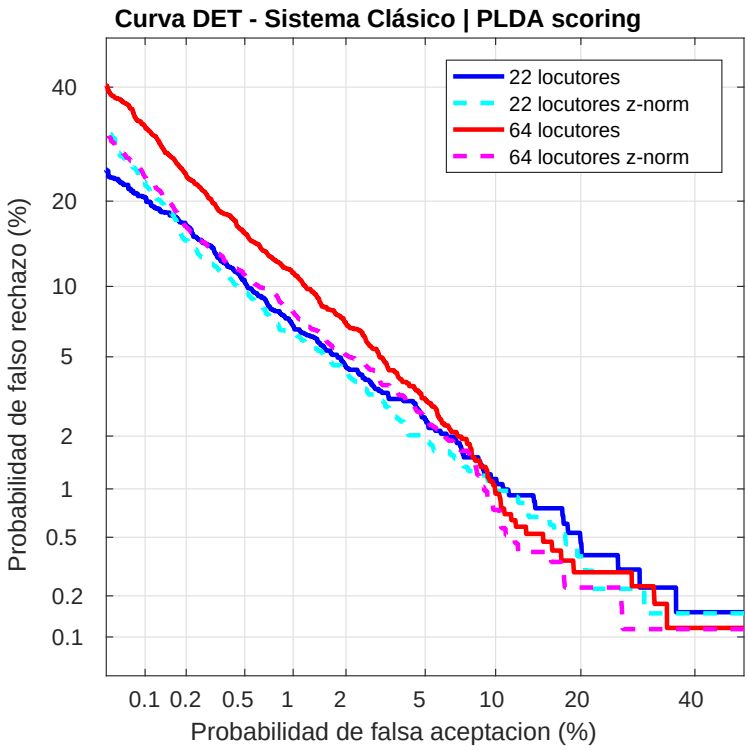

Figura 6.1: Curva DET del sistema *baseline*, para las versiones de 22 y 64 locutores con y sin z-norm.

A partir de los resultados obtenidos se puede determinar:

- Mejora sustancial del rendmiento con el uso del método PLDA frente a *cosine scoring* y LDA. Principalmente se observa mayor *gap* cuando el número y duración de las

locuciones de entrenamiento es muy bajo (sistema 64 locutores). Esto permite identificar PLDA como un método que permite modelar la variabilidad inter e intra-locutor, de manera suficientmente robusta en locuciones relativamente cortas (aproximadamente 5-10 segundos) [Kanagasundaram et al., 2011]

- Cierta mejora en el rendimiento con el uso de normalización de *scores* z-norm. Esto advierte de un pequeño desalineamiento de *scores* que afecta mínimamente al sistema por su umbral independiente de locutor, y es corregido parcialmente con z-norm.

Por tanto, como punto de referencia se obtiene un sistema con error 3.13 % de EER en su versión de 22 locutores (versión optimista, con más datos de entrenamiento) y 3.59 % de EER en su versión de 64 locutores (versión pesimista, con menos datos de entrenamiento).

6.2. Sistema duraciones cortas

Como se comenta en el Capítulo 1 los sistemas de detección de hablantes en locuciones cortas deben trabajar con flujos continuos de audio. Es decir, no se realizan comparaciones entre dos locuciones claramente determinadas, sino que deben ser capaces de realizar una estimación en cortos periodos temporales sobre la presencia de locutor y la identificación del mismo. Por tanto, existe la posibilidad de realizar comparaciones entre i-vectors que provengan de distintas fuentes: voz, música, ruido, solapamiento, transiciones, etc.

Para medir el rendimiento del sistema bajo este tipo de condiciones, se proponen una serie de experimentos detallados a continuación. Cada uno ello contempla una serie de mejoras propuestas en la descripción del sistema (Capitulo 4) que tratan de lidiar con la problemática intrínseca de esta tarea.

Dichos experimentos se enmarcan bajo el protocolo de evaluacion definido en el Capítulo 5. Principalmente, debe tenerse en cuenta la supresión de fronteras entre transiciones voz/no-voz, tal y como se detalla en la sección 5.4.1. Esto permite eliminar de la evaluación los resultados que provengan de i-vectors pertenecientes a voz y no-voz de forma conjunta.

6.2.1. Extracción de i-vectors de corta duración con aumento de resolución.

En esta seccion se evaluará el rendimiento del sistema detallado en la Sección 4.2. Donde se trabaja con un flujo continuo de audio, realizado una extracción de i-vectors con duración de segmentos de 5 segundos y un solapamiento entre ellos de 1 segundo. Los experimentos realizados pretenden evaluar el sistema en un entorno lo más realista posible, donde exista todo tipo de audio presente en programas *broadcast* (entrevistas, mitines, noticiaron, tertulias, etc.)

Para ello, se evalúa el sistema con sus dos versiones de entrenamiento (22 y 64 locutores) y se tomarán como segmentos de evaluación todo el audio disponible en Audias-Radio-2015 (exceptuando segmentos de entrenamiento y fronteras de transición). Realizando un total de 90000 *trials*.

La tabla 6.2 muestra los resultados obtenidos para cada una de las versiones, los distintos métodos de puntuación y normalización de *scores* z-norm:

Sistema		EER(%)		
Locutores	Normalización	Cosine	LDA	PLDA
22	-	2.56	2.18	**1.46**
22	z-norm	2.91	2.82	1.99
64	-	6.53	5.99	3.95
64	z-norm	5.15	5.18	**3.63**

Tabla 6.2: Rendimiento del sistema de locuciones cortas (raw) en EER(%).

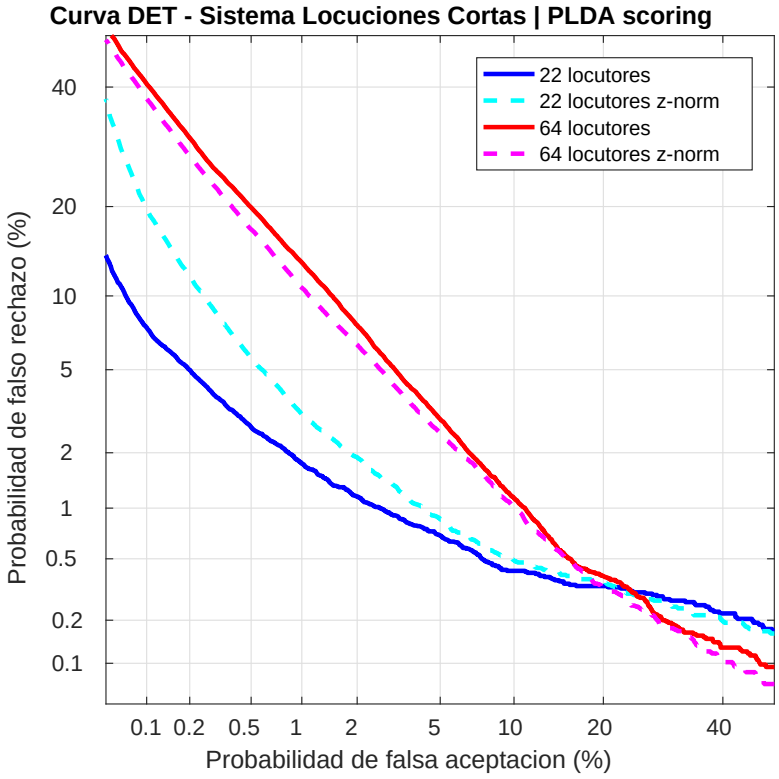

Figura 6.2: Curva DET del sistema de locuciones cortas (raw), para las versiones de 22 y 64 locutores con y sin z-norm.

A la vista de los resultados obtenidos, se vuelve a confirmar el hecho de un mejor rendimiento aplicando PLDA en lugar de LDA o *cosine scoring*. Además, los resultados en cuando al uso de una etapa de post-procesado de *scores*, indican una mejora mínima en el sistema de 64 locutores y un empeoramiento mínimo para 22 locutores respecto a no aplicar normalización.

Además, cabe destacar una diferencia importante entre estos resultados y los obtenidos en el sistema *baseline*:

- Para el sistema optimista (22 locutores), el rendimiento es altamente superior al obtenido anteriormente en el sistema clásico. Esto se debe principalmente a la supresión de fronteras en el protocolo evaluación de resultados (esto no es aplicado en la evaluación del sistema clásico). Unido a un entrenamiento robusto, que permite mantener un rendimiento adecuado incluso con extracción de i-vectors de corta duración.

- Para el sistema pesimista (64 locutores), el rendimiento se mantiene similar al obtenido en el sistema *baseline*. Una primera aproximación indica un peor rendimiento debido a los pocos datos de entrenamiento, unidos a la extracción de i-vectors de evaluación de corta duración. Este hecho se ve compensado por la supresión de fronteras y extracción de i-vectors de *test* y *train* con la misma duración.

6.2.2. Promedidado i-vectors

Esta sección presenta los resultados obtenidos para la propuesta de promediado de i-vectors. Este sistema pretende hacer uso de la información redundante presente en i-vectors contiguos para tratar de realiza una mejor estimación de los mismos. Por tanto, se utilizarán i-vectors contiguos para realizar un promediado que en cierta manera mejore la estimación de los mismos. Los resultados se muestran para dos tipos de promediado, -lineal y hamming-, permitiendo proporcionar diferentes pesos dependiendo de la distancia al i-vector central, ver sección 4.2.2..

Cabe destacar un aumento de la región de frontera dependiente de la cantidad de i-vectors contiguos promediados (como se indica en el Capítulo 5). Por tanto, las regiones frontera abarcarán todas aquellas zonas susceptibles de utilizar i-vectors que pertenezcan a segmento de solapamiento voz/no-voz y por tanto, serán descartadas en la evaluación del rendimiento.

Sistema			EER(%)		
Locutores	Normalización	Ventana promedio	Cosine	LDA	PLDA
22	-	raw*	2.54	2.17	1.47
22	-	3-lineal	2.44	1.88	**1.30**
22	-	3-hamming	2.48	1.91	1.37
22	z-norm	raw*	2.90	2.80	1.97
22	z-norm	3-lineal	2.86	2.49	1.76
22	z-norm	3-hamming	2.89	2.88	1.89
64	-	raw*	6.57	5.96	3.92
64	-	3-lineal	5.85	5.10	3.34
64	-	3-hamming	6.32	5.55	3.52
64	z-norm	raw*	5.18	5.16	3.61
64	z-norm	3-lineal	4.53	4.39	**3.05**
64	z-norm	3-hamming	5.04	4.42	3.38

Tabla 6.3: Rendimiento del sistema promedidado de 3-ivectors. Para raw* se aplica el mismo protocolo de supresión de frontera que en el tipo de promediado con el que se compara.

A partir de los resultados obtenidos, tablas 6.3-6.5 y figuras 6.3-6.4, se observa una ligera mejora en el rendimiento global del sistema cuando se aplica algún tipo promediado de i-vectors. En particular, utilizando un promediado lineal (misma ponderación para todos los i-vectors) surge el mejor rendimiento observado. Previsiblemente, esto sucede debido al hecho de utilizar información redundante presente en i-vectors solapados, permitiendo realizar una mejor estimación de los mismos. Además, aplicando normalización de scores z-norm, se mantiene la misma dinámica vista sin promediado de i-vectors, dónde se produce mejora para la versión pesimista de 64 locutores y empeora para la versión optimista de 22 locutores.

Por tanto, en líneas generales se obtiene en decrecimiento del EER en 0.10 %-0.60 % para cualquier configuración en promediado de i-vectors, respecto del sistema base de locuciones cortas (raw).

Sistema			EER(%)		
Locutores	Normalización	Ventana promedio	Cosine	LDA	PLDA
22	-	raw*	2.26	1.99	1.28
22	-	5-lineal	2.17	1.50	**1.08**
22	-	5-hamming	2.21	1.72	1.14
22	z-norm	raw*	2.70	2.61	1.85
22	z-norm	5-lineal	2.64	2.07	1.48
22	z-norm	5-hamming	2.72	2.39	1.74
64	-	raw*	5.98	5.63	3.85
64	-	5-lineal	5.20	4.30	2.81
64	-	5-hamming	5.43	5.40	3.55
64	z-norm	raw*	4.91	5.01	3.59
64	z-norm	5-lineal	4.00	3.72	**2.53**
64	z-norm	5-hamming	4.68	4.97	2.92

Tabla 6.4: Rendimiento del sistema promediado de 5-ivectors. Para raw* se aplica el mismo protocolo de supresión de frontera que en el tipo de promediado con el que se compara.

Sistema			EER(%)		
Locutores	Normalización	Ventana promedio	Cosine	LDA	PLDA
22	-	raw*	2.01	1.82	1.15
22	-	7-lineal	1.88	1.24	**0.94**
22	-	7-hamming	1.91	1.68	1.10
22	z-norm	raw*	2.48	2.46	1.71
22	z-norm	7-lineal	2.44	1.73	1.31
22	z-norm	7-hamming	2.41	2.22	1.50
64	-	raw*	6.09	5.69	3.71
64	-	7-lineal	4.63	3.61	2.42
64	-	7-hamming	5.79	4.84	3.53
64	z-norm	raw*	4.74	4.83	3.47
64	z-norm	7-lineal	3.57	3.16	**2.09**
64	z-norm	7-hamming	3.97	4.15	3.22

Tabla 6.5: Sistema promediado de 7-ivectors. Para raw* se aplica el mismo protocolo de supresión de frontera que en el tipo de promediado con el que se compara.

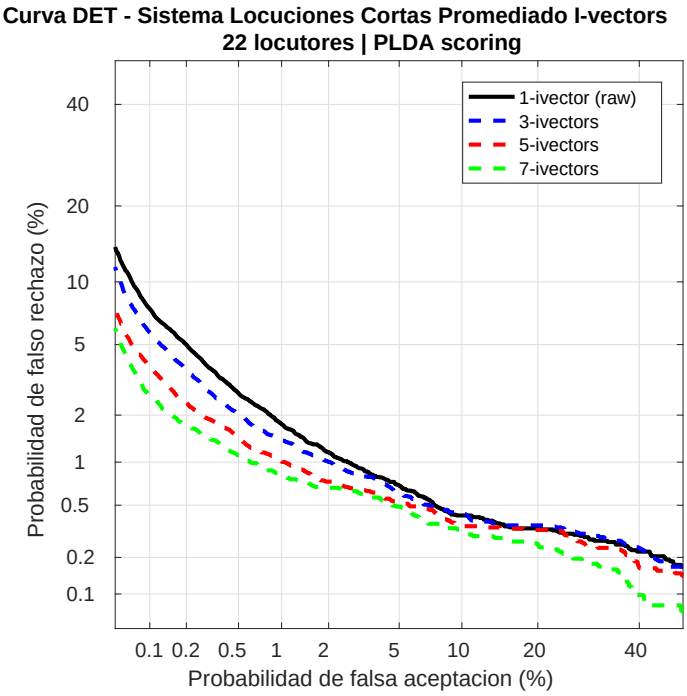

Figura 6.3: Curva DET del sistema de locuciones cortas con promediado de i-vectors para 22 locutores.

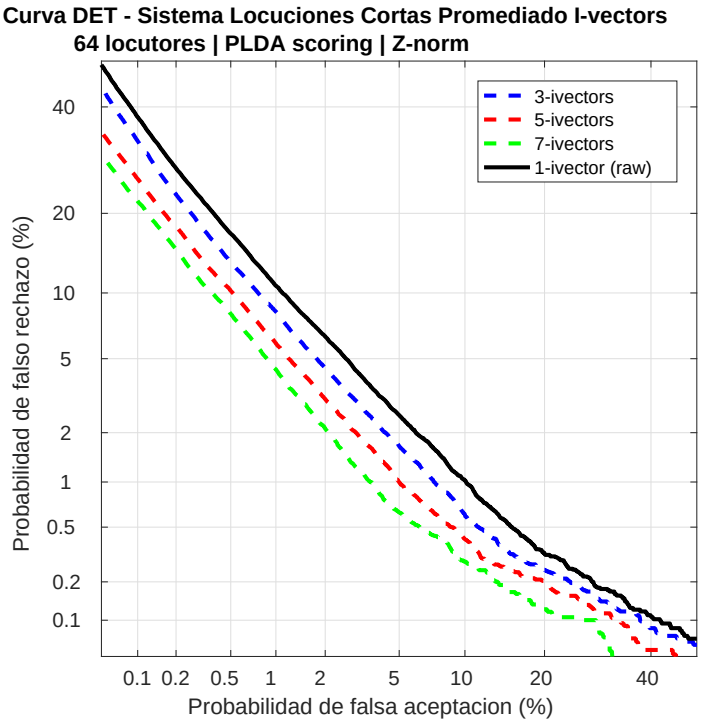

Figura 6.4: Curva DET del sistema de locuciones cortas con promediado de i-vectors para 64 locutores, con z-norm.

6.2.3. Promediado scores

Siguiendo el mismo procedimiento descrito en el apartado anterior, se detallan los resultados obtenidos para el promediado en el dominio de *scores*. Este promediado pretende compensar cierta variabilidad presente entre *scores* contiguos, por tanto, aplica un suavizado sobre los *scores* de salida eliminando de esta manera cierto *scores* espúreos sin significado contextual respecto a los *scores* de su vecindario. Para ello, se han aplicado dos tipos de ventanas en las pruebas: lineal y Hamming. Con tres configuraciones diferentes en cuanto al número de *scores* promediados: 3, 5 y 7. Además, se mantienen las pruebas con el método de normalización z-norm.

En este caso, de igual manera que el apartado anterior se eliminan de la evaluación aquellos *scores* que contienen información de segmentos de transición.

Sistema			EER(%)		
Locutores	Normalización	Ventana promedio	Cosine	LDA	PLDA
22	-	raw*	2.54	2.17	1.47
22	-	3-lineal	2.29	1.91	**1.32**
22	-	3-hamming	2.48	2.11	1.42
22	z-norm	raw*	2.90	2.80	1.97
22	z-norm	3-lineal	2.71	2.55	1.78
22	z-norm	3-hamming	2.88	2.73	1.92
64	-	raw*	6.57	5.96	3.92
64	-	3-lineal	5.98	5.34	3.54
64	-	3-hamming	5.79	5.39	3.83
64	z-norm	raw*	5.18	5.16	3.61
64	z-norm	3-lineal	4.46	4.59	**3.21**
64	z-norm	3-hamming	4.75	4.42	3.52

Tabla 6.6: Sistema promediado de 3-*scores*. Para raw* se aplica el mismo protocolo de supresión de frontera que en el tipo de promediado con el que se compara.

Sistema			EER(%)		
Locutores	Normalización	Ventana promedio	Cosine	LDA	PLDA
22	-	raw*	2.26	1.99	1.28
22	-	5-lineal	1.83	1.55	**1.00**
22	-	5-hamming	2.00	1.70	1.14
22	z-norm	raw*	2.70	2.61	1.85
22	z-norm	5-lineal	2.32	2.18	1.52
22	z-norm	5-hamming	2.49	2.35	1.69
64	-	raw*	5.98	5.63	3.85
64	-	5-lineal	5.30	4.72	3.00
64	-	5-hamming	5.45	4.99	3.38
64	z-norm	raw*	4.91	5.01	3.59
64	z-norm	5-lineal	3.99	3.99	**2.73**
64	z-norm	5-hamming	4.41	4.39	3.15

Tabla 6.7: Sistema promediado de 5-*scores*. Para raw* se aplica el mismo protocolo de supresión de frontera que en el tipo de promediado con el que se compara.

Sistema			EER(%)		
Locutores	Normalización	Ventana promedio	Cosine	LDA	PLDA
22	-	raw*	2.01	1.82	1.15
22	-	7-lineal	1.47	1.29	**0.78**
22	-	7-hamming	1.68	1.46	0.96
22	z-norm	raw*	2.48	2.46	1.71
22	z-norm	7-lineal	1.95	1.83	1.23
22	z-norm	7-hamming	2.24	2.10	1.46
64	-	raw*	6.09	5.69	3.71
64	-	7-lineal	4.61	4.07	2.56
64	-	7-hamming	5.20	4.71	3.02
64	z-norm	raw*	4.74	4.83	3.47
64	z-norm	7-lineal	3.51	3.42	**2.33**
64	z-norm	7-hamming	4.25	3.94	2.79

Tabla 6.8: Sistema promediado de 7-*scores*. Para raw* se aplica el mismo protocolo de supresión de frontera que en el tipo de promediado con el que se compara.

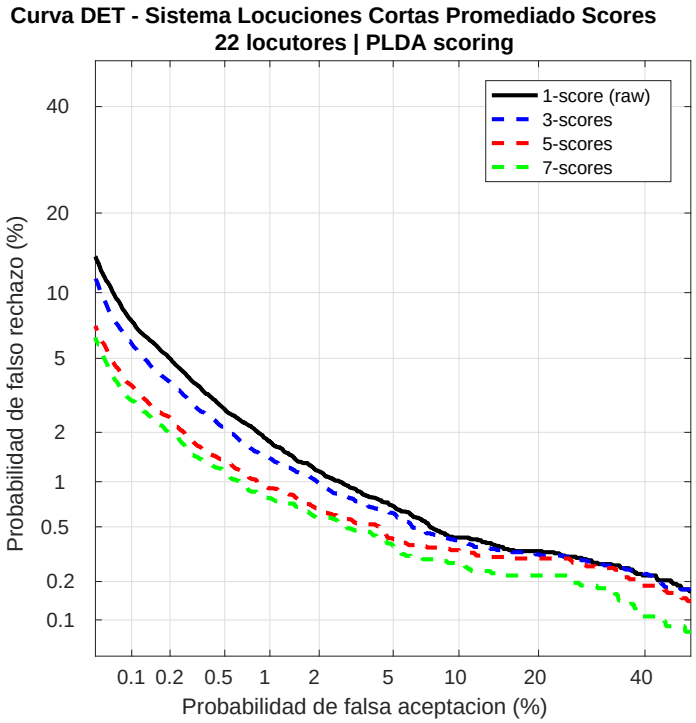

Figura 6.5: Curva DET del sistema de locuciones cortas con promediado de *scores* para 22 locutores.

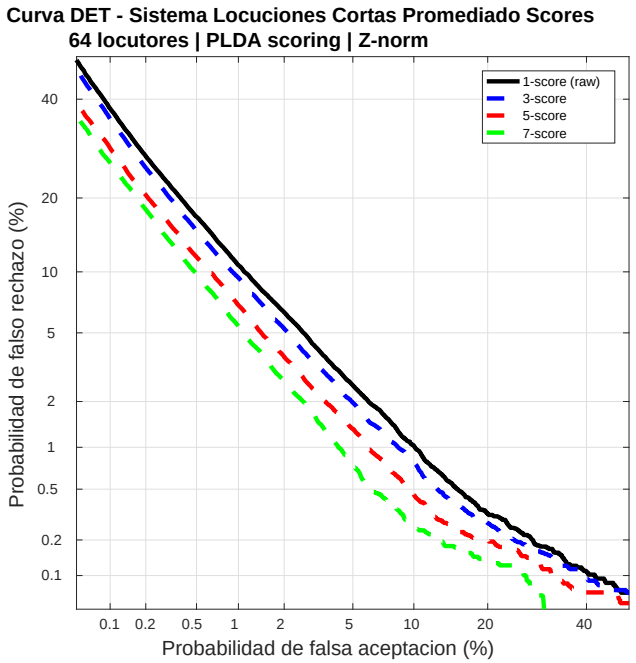

Figura 6.6: Curva DET del sistema de locuciones cortas con promediado de *scores* para 64 locutores con z-norm.

A partir de los resultados obtenidos, Tablas 6.6-6.8 y Figuras 6.5-6.6 y de forma similar a los resultados obtenidos para el promediado de i-vectors, se presenta una ligera mejora en el rendimiento del sistema para todas las configuraciones posibles. Por tanto, el uso de información contextual y redundante presente en scores provenientes de i-vectors cercanos, permite una mejora del sistema respecto al sistema básico de locuciones cortas (raw). Si bien es cierto, que presenta un rendimiento muy similar a los obtenidos durante los experimentos con promediado en el dominio de los i-vectors. Se puede determinar una técnica con una mejora sustancial en el decrecimiento del EER en torno a un 0.10 %-0.60 % respecto a un sistema sin post-procesado de *scores*. Además, permite ventajas en términos de rendimiento computacional frente al promediado de i-vectors Donde se pasa de realizar 600 operaciones de promediado (1 por dimensión del i-vector) para el caso del promediado i-vector, frente a 22 ó 64 operaciones (1 *score* por locutor) en el caso del promediado de *scores*.

6.2.4. Concatenación locuciones de entrenamiento

En esta sección se evaluará el rendimiento del sistema modificando el uso de las locuciones de entrenamiento utilizadas hasta ahora. Para ello, se supone el hecho de que locuciones de mayor duración dan lugar a i-vectors que representan mejor la información de manera robusta. En cambio, locuciones cortas, permiten extraer versiones ruidosas" de los mismos. Teniendo en cuenta el entorno experimental utilizado en este proyecto donde mayoritariamente se tienen locuciones cortas, se propone este experimento, como método de validación sobre la manera más eficiente de extraer los i-vectors de entrenamiento. Para ello, se realizará una comparativa del sistema base de locuciones cortas en dos opciones:

- I-vector media: Se realiza un promediado de todos los i-vectors de entrenamiento para cada locutor (hasta ahora se había realizado de esta manera).
- I-vector concatenación: Se toman todas las locuciones como una única (concatenación) y se extrae un único i-vector de ella.

Sistema		EER(%)		
Locutores	Normalización	Cosine	LDA	PLDA
22-media	-	2.56	2.18	1.46
22-concat	-	2.50	2.25	**1.36**
22-media	z-norm	2.91	2.82	1.99
22-concat	z-norm	2.79	2.84	1.88
64-media	-	6.53	5.99	3.95
64-concat	-	7.53	6.66	4.07
64-media	z-norm	5.15	5.18	**3.63**
64-concat	z-norm	5.66	5.64	3.81

Tabla 6.9: Rendimiento del sistema con media de i-vectors de entrenamiento y concatenación de locuciones de entrenamiento, con y sin z-norm.

A la vista de los resultados obtenidos, se observa una mínima mejora muy poco significativa en el sistema optimista de 22 locutores (mayor duración en locuciones de entrenamiento), quizá proveniente de i-vectors de entrenamiento más robustos, debido a la concatenación locuciones. Por otro lado, se observa un mínimo empeoramiento para el sistema de 64 locutores (entrenamiento con 3 segmentos de 5 segundos), posiblemente debido a una mayor relación

entre las duraciones coincidentes de los i-vectors de entrenamiento y evaluación (5 segundos entrenamiento vs 5 segundos evaluación), frente a la la concatenación, que da lugar a i-vectors estimados a partir de diferentes duraciones (15 segundos entrenamiento vs 5 segundos test).

6.3. Sistema Adaptación de Dominio

En esta sección se mostrarán los experimentos realizados y resultados obtenidos sobre las mejoras propuestas para el sistema en la tarea específica de Adaptación de Dominio (de audio telefónico/inglés a audio microfónico/castellano). Para ello, en primer lugar se propone un estudio sobre como afecta el entrenamiento de los distintos hiper-parámetros con las distintas opciones de audio (bases de datos) de las que disponemos. Su objetivo es determinar las etapas críticas en la mejora de rendimiento, para una adaptación del sistema a un entorno real, del que se disponen pocos datos de entrenamiento.

En segundo término, se mostrarán los resultados obtenidos para las distintas técnicas implementadas para realizar la adaptación de dominio. Dichas técnicas incluyen métodos supervisados y no supervisados, que harán uso de datos etiquetados manualmente y de técnicas de *clustering* para la realización de etiquetado automático de los datos.

Para los siguientes experimentos se tomará un conjunto de 64 locutores *target* (mismo sistema pesimista descrito anteriormente), utilizando 3 locuciones de 5 segundos para el entrenamiento de sus respectivos modelos.

Además, dichos experimentos se enmarcan bajo el protocolo de evaluación definido en el Capítulo 4. Debiendo tenerse en cuenta la eliminación de fronteras entre transiciones voz/no-voz, tal y como se detalla en la Sección 5.4.1.

6.3.1. Entrenamiento de hiper-parámetros.

Con el objetivo de determinar el comportamiento de cada uno de los bloques presentes en el sistema de reconocimiento de locutor, se realizan los siguientes experimentos en el marco comparativo de cada una de las etapas. Su principal objetivo es tratar de determinar cuales son las etapas críticas que modifican en mayor medida el rendimiento del sistema, para posteriomente actuar sobre ellas con técnicas de compensación de dominio.

Los experimentos se realizan sobre la misma configuración del sistema, donde varían los datos de entrenamiento en cada una de las etapas. Para ello se tomarán los distintos conjuntos de datos compuestos de la siguiente manera:

- OUT: Corresponde con datos *out-domain*: Development Dataset (SRE+SWBD) etiquetado 5500 horas).
- IN: Corresponde con datos *in-domain*: Audias-Radio-2017 (excepto audio de entrenamiento de 64 locutores). Etiquetados para etapas supervisadas (PLDA), (25 horas) y no etiquetados para etapas no supervisadas (UBM, T, W), (250 horas).
- OUT+IN: Corresponde con la combinación de datos *out-domain* e *in-domain*.

UBM,T	m, W	PLDA(Σ, Φ)	EER(%)
OUT	OUT	OUT	3.63
OUT	OUT	OUT+IN	7.53
OUT	OUT	IN	11.06
OUT	OUT+IN	OUT	4.69
OUT	OUT+IN	OUT+IN	9.08
OUT	OUT+IN	IN	17.50
OUT	IN	OUT	5.54
OUT	IN	OUT+IN	8.19
OUT	IN	IN	8.68
OUT+IN	OUT	OUT	4.02
OUT+IN	OUT	OUT+IN	5.32
OUT+IN	OUT	IN	16.33
OUT+IN	OUT+IN	OUT	4.88
OUT+IN	OUT+IN	OUT+IN	7.45
OUT+IN	OUT+IN	IN	9.16
OUT+IN	IN	OUT	6.51
OUT+IN	IN	OUT+IN	8.12
OUT+IN	IN	IN	8.82
IN	OUT	OUT	11.56
IN	OUT	OUT+IN	13.11
IN	OUT	IN	10.73
IN	OUT+IN	OUT	11.50
IN	OUT+IN	OUT+IN	11.66
IN	OUT+IN	IN	10.02
IN	IN	OUT	11.14
IN	IN	OUT+IN	11.04
IN	IN	IN	9.94

Tabla 6.10: Resultados en función de entrenamiento de hiper-parámetros para diferentes conjuntos de datos *in-domain/out-domain*.

A partir de los resultados obtenidos, se pueden realizar varias apreciaciones:

- Existe una importante diferencia entre el sistema entrenado únicamente con gran cantidad de datos *out-domain* y el sistema entrenado únicamente con una pequeña cantidad de datos *in-domain*. Esto nos demuestra la importancia de la cantidad de datos en el rendimiento de este tipo de sistemas de reconocimiento automático de locutor. Dónde en este caso, prima la cantidad de datos disponible al dominio específico de donde provengan esos datos.
- Se observa la etapa de PLDA (fase de puntuación entrenada de manera supervisada) como la etapa con mayor impacto en el rendimiento del sistema. Es por ello, que se utilizará en los siguientes experimentos de adaptación de dominio.

6.3.2. Adaptación de Dominio Supervisada

En esta sección se mostrarán los resultados obtenidos para la adaptación de dominio de forma supervisada. Para ello, se realizarán pruebas aplicando la técnica de interpolación de

matrices de covarianza PLDA, dicha técnica se detalla en el Capítulo 4. Con el objetivo de determinar el comportamiento del sistema según el grado de adaptación, se procede a realiza un barrido del coeficiente de adaptación, α. Donde α tomará valores entre 0 y 1, siendo $\alpha = 0$ entrenamiento únicamente con datos *out-domain* y $\alpha = 1$ corresponde a un entrenamiento completamente con datos in-domain. Los conjuntos de datos se corresponden con:

- Out-Domain: Datos de desarrollo (SRE+SWBD) (5500 horas).
- In-Domain: Audias-Radio-2017 Etiquetada (excepto audio de entrenamiento para 64 locutores) (25 horas).

A la vista de los resultados obtenidos en la Figura 6.8, se comprueba una pérdida del rendimiento del sistema según aumenta el coeficiente de adaptación. Por tanto, a medida que aumenta la importancia de los datos *in-domain* en el entrenamiento PLDA aumenta el error cometido. Esto indica una adaptación totalmente inefectiva, que puede deberse a una cantidad de datos insuficientes de *in-domain*, lo que se traduciría en una incorrecta estimación de las matrices de covarianza inter-clase e intra-clase, afectando negativamente al rendimiento del sistema. Por tanto, para el entorno experimental desarrolado, la naturaleza y cantidad de los audios disponibles, no se obtiene mejora en el rendimiento. De forma contraria a los resultados obtenidos en [Garcia-Romero and McCree, 2014] con este tipo de técnica.

6.3.3. Adaptación de Dominio No Supervisada

En esta sección muestra los resultados obtenidos para la adaptación de dominio de forma no supervisada. Con el objetivo de aumentar el número de datos de entrenamiento disponibles para aplicar la técnica de interpolación de matrices de covarianza PLDA (detallada en el Capítulo 4). Para ello, se propone un algoritmo de *clustering* como AHC (*Agglomerative Hierarchical Clustering*) que permita realizar un etiquetado "automático" sobre qué locuciones pertenecen a un mismo locutor. Esto permite aumentar en número de datos de entrenamiento y corroborar si la técnica aplicada en la sección anterior no implica mejoras en los resultados, justamente, por la falta de datos. Los conjuntos de datos se corresponden con:

- Out-Domain: Datos de desarrollo (SRE+SWBD) (5500 horas).
- In-Domain: Audias-Radio-2017 No Etiquetada (excepto audio train 64 locutores) (250 horas).

Parámetros AHC

La realización del *clustering* se realiza a nivel de i-vector, a partir del algoritmo AHC basado en distancia coseno para la unión de *clusters* y un criterio de parada a partir de un valor de corte. Este valor de corte (cutoff) se determina a partir del subconjunto etiquetado de datos *in-domain* (Audias-Radio-2015 etiquetada, 25 horas). Para ello se realiza un estudio del grado de impureza por clases y *clusters*. Determinando el valor óptimo de *cutoff* donde ambas son iguales:

Esto nos permite fijar el valor de cutoff en 0.77, que se utilizará como criterio de parada en el *clustering* de las 250 horas de Audias-Radio-2015 sin etiquetar.

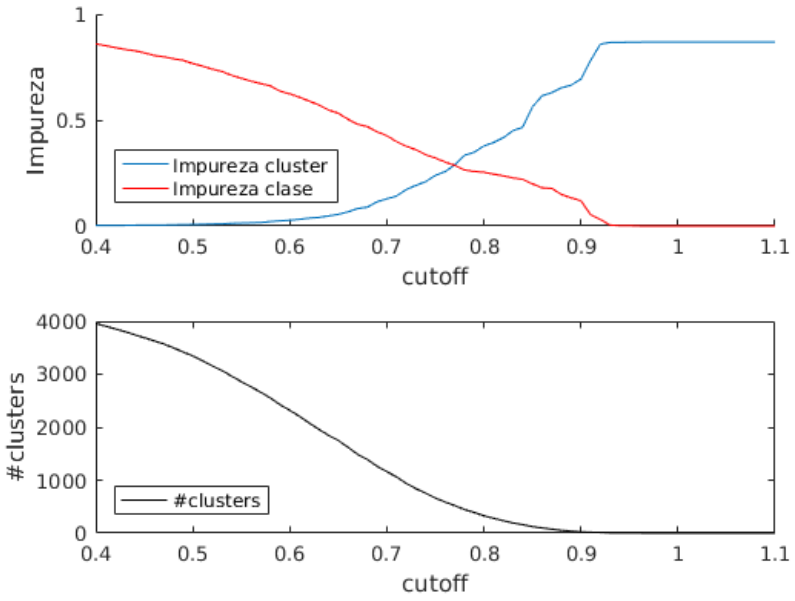

Figura 6.7: Representación de impureza de *clusters* y clases, según el valor del término *cuttoff* (arriba). Representación del número de *clusters* dependiente del valor de *cutoff*

AHC + Adaptación de matrices de covarianza PLDA

A partir del criterio de detención (*cutoff*) calculado anteriormente y tomando como medida de distancia coseno. Se procede al *clustering* de los i-vectors pertenecientes a la parte sin etiquetar de los datos in-domain. Esto permite obtener un etiquetado "automático" no supervisado, de gran cantidad de audio.

Aplicando las mismas pruebas desarrolladas en la Sección 4.3.2, pero en esta ocasión haciendo uso de un mayor número de datos etiquetados por locutor mediante *clustering*, se obtienen los siguientes resultados, ver Figura 6.8.

A partir de los resultados obtenidos se observa una mejora significativa en el rendimiento del sistema cuando se dispone de mayor cantidad de datos. Esto comprueba que el rendimiento del sistema (aunque en este caso no consigue mejorar) es susceptible de mejorar cuando la cantidad de datos es suficientemente alta para estimar las matrices de covarianza de manera conveniente. Por tanto, incluso con el previsible error cometido por el *clustering* de datos, un aumento de la cantidad de datos disponibles para el entrenamiento, permite aumentar el rendimiento alcanzado por el sistema.

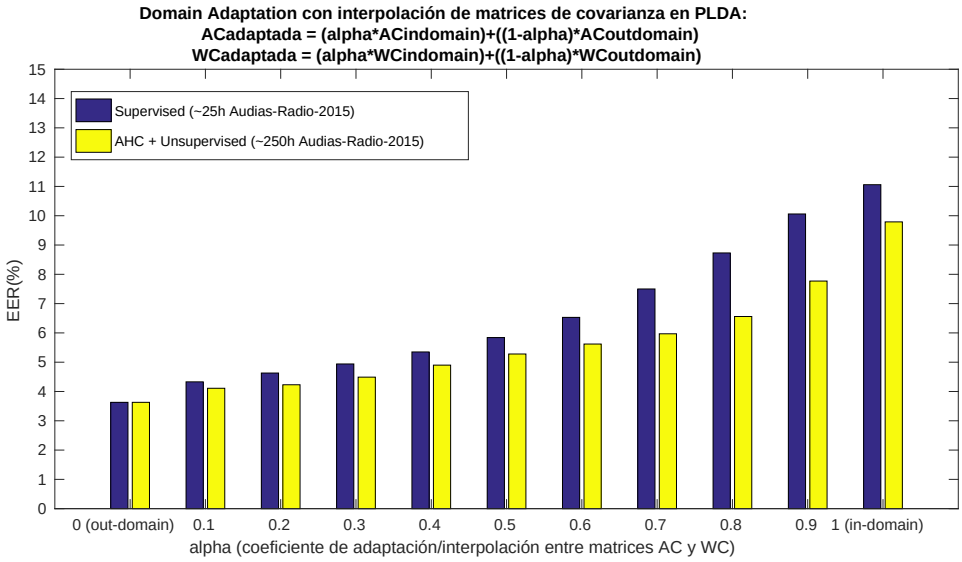

Figura 6.8: Comparativa de resultados obtenidos para interpolación de matrices de covarianza PLDA, con datos supervisados y AHC + datos sin supervisar.

Capítulo 7

Conclusiones y trabajo futuro

7.1. Conclusiones

Durante el presente Trabajo de Fin de Máster se ha estudiado, desarrollado y optimizado un sistema de detección de hablantes sobre un entorno de locuciones cortas y audio *broadcast*. Este marco de trabajo conlleva una alta variabilidad y degradación en el rendimiento del sistema, que se ha tratado de paliar mediante el uso de técnicas aplicables a locuciones cortas y adaptación de dominio.

Para ello, se ha realizado pruebas de rendimiento para cada una de las mejoras propuestas. Que permiten extraer las siguientes conclusiones:

- Los sistemas de detección de hablantes en locuciones cortas y que por tanto, trabajan con flujos continuos de audio (EER 3.63 %) presentan un rendimiento muy similar a los sistemas "clásicos" de reconocimiento de locutor (locuciones pre-segmentadas),(EER 3.59 %). Esto presenta un escenario altamente válido para el uso de este tipo de sistemas en un entorno de detección en tiempo real.
- Según los resultados obtenidos para las mejoras sobre locuciones cortas:
  - El uso de técnicas de promediado de i-vectors para las diferentes configuraciones propuestas implican una pequeña mejora del rendimiento. Por otro lado, el coste computacional que implica dicho promediado también aumenta. Esto presenta un compromiso entre velocidad de computación y mejora en el rendimiento.
  - En el caso de las técnicas de promediado aplicadas a nivel de *score*, se produce un efecto muy similar, donde se produce una mejora mínimamente significativa. Pero, en este caso el rendimiento computacional se reduce, respecto del promediado de i-vectors, debido a realizar el promediado en un espacio de menor dimensión.
  - A partir de las pruebas realizadas sobre la concatenación de locuciones de entrenamiento, con el objetivo de obtener un único i-vector, frente a las técnicas clásicas de extracción de i-vectors individuales y obtener su media (i-vector medio). Se obtiene un rendimiento similar en ambos casos, por tanto, no representa una mejora significativa.
- Según los resultados obtenidos para las mejoras propuestas en adaptación de dominio:
  - El estudio realizado sobre el entrenamiento de hiper-parámetros con diferentes conjuntos de datos, corrobora los estudios presentados en [Garcia-Romero and McCree, 2014], dónde se indica PLDA cómo etapa crítica sobre el rendimiento del sistema.

- Las pruebas realizadas sobre adaptación de dominio a partir de datos etiquetados (supervised domain adaptation), no presentan mejoras en el rendimiento del sistema. Esto queda en contraposición con, [Garcia-Romero and McCree, 2014], donde se presentan mejorar sustanciales aplicando técnicas de adaptación de dominio. Esto se debe a la diferencia entre la cantidad de datos utilizados en ambos experimentos para la adaptación de dominios y la alta variabilidad entre las distintas bases de datos utilizadas en este proyecto. Donde [Garcia-Romero and McCree, 2014] realiza adaptación entre dominios muy similares (mismo idioma y condiciones de grabación), frente a los dominios utilizados en este proyecto (diferentes idiomas y distintas condiciones de grabación).
- Para comprobar la diferencia de resultados obtenidos con los resultados esperados. A partir de los experimentos de adaptación de dominio con datos no etiquetado se observa una clara mejora cuando la cantidad de datos aumenta. Esto permite afirmar que existe un compromiso claro entre el rendimiento del sistema y la cantidad de datos utilizados durante su desarrollo. Por tanto, las mejoras que otorgan las técnicas de *clustering* sobre este tipo de adaptaciones tienden a realizar una mejora en el rendimiento y en teoría permitirían batir el rendimiento del sistema cuando la cantidad de datos utilizados sea suficientemente grande.

Todo ello, ha permitido construir un sistema end-to-end de detección de locutores que trabaja con flujos continuos de audio (programas radiofónicos) con el objetivo de detectar aquellos segmentos donde se produce la aparición de locutores dados de alta en el sistema. Obteniendo un rendimiento máximo del sistema para 22 locutores a detectar del 1.46 % (EER) y para 64 locutores del 3.63 % (ver Capítulo 5).

7.2. Trabajo futuro

A partir del buen rendimiento del sistema bajo condiciones de alta variabilidad y pocos datos disponibles para el dominio de la aplicación. Se proponen una serie de líneas de trabajo futuro, con el objetivo de explorar nuevas técnicas que permitan mejorar el rendimiento del sistema

- Se propone el desarrollo e implementación de nuevas técnicas presentes en el estado del arte, ver sección 3.2.1. Con el objetivo de comprobar el correcto funcionamiento de todas ellas sobre esta problemática.
- Estudio pormenorizado del rendimiento del sistema para diferentes condiciones en cuanto a cantidad de datos y dominios diferentes. Todo ello, se debe realizar sobre un entorno experimental adecuado y convenientemente desarrollado para tal efecto.
- Evaluar el rendimiento del sistema con datos provenientes de distintos dominios y diferentes entornos, con el objetivo de valorar si es posible la extrapolación de estos resultados para cualquier aplicación desarrollable.
- Implementación del sistema de detección para realizar el procesado de i-vectors y *scoring* en tiempo real. Para ello se deberá utilizar procesado de datos *on-line* y alta capacidad de cómputo.

Glosario de acrónimos

- **CMVN**: Cepstral Mean and Variance Normalization (normalización de la media y varianza cepstral). Técnica utilizada en el dominio de los coeficientes cepstrales con el objetivo de reducir los efectos del canal de transmisión.
- **DFT**: Discrete Fourier Transform (transformada discreta de Fourier). Función de transformación de dominio temporal a dominio espectral en tiempo discreto.
- **DET**: Detection Error Trade-off (compensación por error de detección). Presenta de forma gráfica el rendimiento de los sistemas de reconocimiento para todo punto de trabajo posible. Es ampliamente utilizada para la evaluación de sistemas.
- **EER**: Equal Error Rate (tasa de igual error). Representa con un valor numérico el rendimiento del sistema, en el punto de trabajo donde el error de falsa aceptación y falso rechazo son iguales.
- **E-M**: Expectation-Maximization. Algoritmo iterativo de estimación y reasignación de parámetros de Gaussianas.
- **FA**: Factor Analysis. Técnica de modelado de variabilidad inter e intra-sesión
- **GMM**: Gaussian Mixture Models (modelos de mezclas de Gaussianas). Técnica de modelado de locutor a partir de sus características y el uso de mezclas de Gaussianas multivariadas.
- **JFA**: Joint Factor Analysis. Técnica de compensación de variabilidad conjuntamente entre inter e intra-sesión.
- **LDA**: Linear Discriminant Analysis (análisis lineal discriminativo). Método aplicado para aumentar la separación de clases durante el scoring.
- **MAP**: Maximum a Posteriori. Técnica de adaptación aplicada sobre un modelo UBM con el objetivo de ajustarlo a un locutor objetivo.
- **MFCC**: Mel Frequency Cepstral Coefficients (coeficientes cepstrales en escala frecuencial Mel). Coeficientes de características extraídos a partir de un análisis auditivo humano.
- **NIST**: National Institute of Standards and Technology (Instituto Nacional de Estándares y Tecnología de EE.UU..)
- **Non-Target**: Muestra que no pertence al modelo o patrón con el que ha de ser siendo comparado.
- **Score**: Puntuación obtenida como medida de similitud entre dos muestras dadas (*trial*)

- **SRE**: Speaker Recognition Evaluations (evaluaciones de reconocimiento de locutor). Evaluaciones propuestas por NIST, para medir el rendimiento de sistemas frente a tareas propuestas.
- **Target**: Muestra que pertenece al modelo o patrón con el que ha de ser comparado.
- **Trial**: Comparativa entre dos muestras dadas, habitualmente, muestras de entrenamiento frente evaluación.
- **TV**: Total Variability. Técnica de compensación de la variabilidad conjunta sobre un subespacio de dimensionalidad reducida.
- **UBM**: Universal Background Model (modelo universal de fondo). Modelo GMM que representa el comportamiento de cualquier locutor.
- **VAD**: Voice Activity Detector (detector de actividad de voz). Sistema que permite descartar aquellas tramas que no sean susceptibles de contener señal de voz.
- **Z-norm**: Zero Normalization. Técnica de normalización de *scores* que permite centrar *trials non-target* en media cero y desviación típica unidad.

Bibliografía

[Adaptation, 2013] Adaptation, J.-C. D. (2013). Johns hopkins university. In *2013 speaker recognition workshop. Available online: http://www. clsp. jhu. edu/workshops/archive/ws13-summerworkshop/groups/spk-13.*

[Aronowitz, 2014] Aronowitz, H. (2014). Inter dataset variability compensation for speaker recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 4002–4006. IEEE.

[Bimbot et al., 2004] Bimbot, F., Bonastre, J.-F., Fredouille, C., Gravier, G., Magrin-Chagnolleau, I., Meignier, S., Merlin, T., Ortega-García, J., Petrovska-Delacrétaz, D., and Reynolds, D. A. (2004). A tutorial on text-independent speaker verification. *EURASIP journal on applied signal processing*, 2004:430–451.

[Campbell, 1997] Campbell, J. P. (1997). Speaker recognition: A tutorial. *Proceedings of the IEEE*, 85(9):1437–1462.

[Dehak et al., 2011] Dehak, N., Kenny, P. J., Dehak, R., Dumouchel, P., and Ouellet, P. (2011). Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):788–798.

[Dempster et al., 1977] Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38.

[Domínguez et al., 2012] Domínguez, J. G., Zazo, R., and González-Rodríguez, J. (2012). On the use of total variability and probabilistic linear discriminant analysis for speaker verification on short utterances. In *Advances in Speech and Language Technologies for Iberian Languages*, pages 11–19. Springer.

[Duda et al., 2001] Duda, R. O., Hart, P. E., Stork, D. G., et al. (2001). Pattern classification. 2nd. *Edition. New York*, page 55.

[Escudero Barrero et al., 2016] Escudero Barrero, Á. et al. (2016). Búsqueda eficiente de audio pregrabado en audio broadcast. B.S. thesis.

[Fant, 1970] Fant, G. (1970). Acoustic theory of speech production (mouton, the hague, 1960). *The closely spaced horizontal lines shown in Fig. 1A are the harmonics of the fundamental frequency of phonation, and are typically revealed in narrowband spectrograms.*

[Fernández Gallego, 2016] Fernández Gallego, M. P. (2016). Mejoras de un sistema de búsquedas en voz y aplicación a detección de menciones en medios de comunicación. B.S. thesis.

[Furui, 1981] Furui, S. (1981). Comparison of speaker recognition methods using statistical features and dynamic features. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(3):342–350.

[García Naranjo et al., 2016] García Naranjo, B. et al. (2016). Segmentación de audio broadcast. B.S. thesis.

[Garcia-Romero and Espy-Wilson, 2011] Garcia-Romero, D. and Espy-Wilson, C. Y. (2011). Analysis of i-vector length normalization in speaker recognition systems. In *Interspeech*, volume 2011, pages 249–252.

[Garcia-Romero and McCree, 2014] Garcia-Romero, D. and McCree, A. (2014). Supervised domain adaptation for i-vector based speaker recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 4047–4051. IEEE.

[Garcia-Romero et al., 2014] Garcia-Romero, D., McCree, A., Shum, S., Brummer, N., and Vaquero, C. (2014). Unsupervised domain adaptation for i-vector speaker recognition. In *Proceedings of Odyssey: The Speaker and Language Recognition Workshop*.

[Hansen and Hasan, 2015] Hansen, J. H. and Hasan, T. (2015). Speaker recognition by machines and humans: A tutorial review. *IEEE Signal processing magazine*, 32(6):74–99.

[Hartigan and Wong, 1979] Hartigan, J. A. and Wong, M. A. (1979). Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108.

[Jain et al., 2000] Jain, A. K., Duin, R. P. W., and Mao, J. (2000). Statistical pattern recognition: A review. *IEEE Transactions on pattern analysis and machine intelligence*, 22(1):4–37.

[Kanagasundaram et al., 2011] Kanagasundaram, A., Vogt, R., Dean, D. B., Sridharan, S., and Mason, M. W. (2011). I-vector based speaker recognition on short utterances. In *Proceedings of the 12th Annual Conference of the International Speech Communication Association*, pages 2341–2344. International Speech Communication Association (ISCA).

[Kenny, 2005] Kenny, P. (2005). Joint factor analysis of speaker and session variability: Theory and algorithms. *CRIM, Montreal,(Report) CRIM-06/08-13*, 14:28–29.

[Kenny et al., 2007] Kenny, P., Boulianne, G., Ouellet, P., and Dumouchel, P. (2007). Joint factor analysis versus eigenchannels in speaker recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(4):1435–1447.

[Kenny and Dumouchel, 2004] Kenny, P. and Dumouchel, P. (2004). Experiments in speaker verification using factor analysis likelihood ratios. In *ODYSSEY04-The Speaker and Language Recognition Workshop*.

[Kinnunen and Li, 2010] Kinnunen, T. and Li, H. (2010). An overview of text-independent speaker recognition: From features to supervectors. *Speech communication*, 52(1):12–40.

[Moon, 1996] Moon, T. K. (1996). The expectation-maximization algorithm. *IEEE Signal processing magazine*, 13(6):47–60.

[Povey et al., 2011] Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., et al. (2011). The kaldi speech recognition toolkit. In *IEEE 2011 workshop on automatic speech recognition and understanding*, number EPFL-CONF-192584. IEEE Signal Processing Society.

[Rabiner and Juang, 1993] Rabiner, L. R. and Juang, B.-H. (1993). Fundamentals of speech recognition.

[Reynolds et al., 2017] Reynolds, D., Singer, E., Sadjadi, S. O., Kheyrkhah, T., Tong, A., Greenberg, C., Mason, L., and Hernandez-Cordero, J. (2017). The 2016 nist speaker recognition evaluation. Technical report, MIT Lincoln Laboratory Lexington United States.

[Reynolds, 2002] Reynolds, D. A. (2002). An overview of automatic speaker recognition technology. In *Acoustics, speech, and signal processing (ICASSP), 2002 IEEE international conference on*, volume 4, pages IV–4072. IEEE.

[Reynolds, 2003] Reynolds, D. A. (2003). Channel robust speaker verification via feature mapping. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, volume 2, pages II–53. IEEE.

[Reynolds et al., 2000] Reynolds, D. A., Quatieri, T. F., and Dunn, R. B. (2000). Speaker verification using adapted gaussian mixture models. *Digital signal processing*, 10(1-3):19–41.

[Sarkar et al., 2012] Sarkar, A. K., Matrouf, D., Bousquet, P. M., and Bonastre, J.-F. (2012). Study of the effect of i-vector modeling on short and mismatch utterance duration for speaker verification. In *Thirteenth Annual Conference of the International Speech Communication Association*.

[Schaefer and Oppenheim, 1989] Schaefer, R. and Oppenheim, A. (1989). Discrete-time signal processing. *D Prentice-Hall*.

[Shum et al., 2014] Shum, S. H., Reynolds, D. A., Garcia-Romero, D., and McCree, A. (2014). Unsupervised clustering approaches for domain adaptation in speaker recognition systems.

[Soriano Morancho et al., 2016] Soriano Morancho, G. et al. (2016). Diarización de locutores en audio broadcast. B.S. thesis.

[Villalba and Lleida, 2014] Villalba, J. and Lleida, E. (2014). Unsupervised adaptation of plda by using variational bayes methods. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 744–748. IEEE.